Prompting Fairness: Learning Prompts for Debiasing Large Language Models

Andrei-Victor Chisca
Computer Science Department
Technical University of Cluj-Napoca
chiscaandrei3@gmail.com

Andrei-Cristian Rad
Computer Science Department
Technical University of Cluj-Napoca
andrei.rad@campus.utcluj.ro

Camelia Lemnaru
Computer Science Department
Technical University of Cluj-Napoca
Camelia.Lemnaru@cs.utcluj.ro

Abstract

Large language models are prone to internalize social biases due to the characteristics of the data used for their self-supervised training scheme. Considering their recent emergence and wide availability to the general public, it is mandatory to identify and alleviate these biases to avoid perpetuating stereotypes towards underrepresented groups. We present a novel prompt-tuning method for reducing biases in encoder models such as BERT or RoBERTa. Unlike other methods, we only train a small set of additional reusable token embeddings that can be concatenated to any input sequence to reduce bias in the outputs. We particularize this method to gender bias by providing a set of templates used for training the prompts[1]. Evaluations on two benchmarks show that our method is on par with the state of the art while having a limited impact on language modeling ability.

1 Introduction

Large language models (LLMs) have claimed state-of-the-art performance on most of the classical natural language processing (NLP) tasks in recent years while facilitating new frontiers in language generation. However, besides being computationally expensive, their performance comes at an additional cost, as they tend to pick up social biases from the vast data required for their pretraining. Consequently, these models can exhibit representational harms, such as disparate system performance, exclusion or stereotyping, or allocation harms, such as discrimination and unequal allocation of resources (Gallegos et al., 2023). With an increased number of use cases and adoption rates, ensuring fairness is becoming more and more critical.

Our work can be summarized by the following key contributions:

- We propose a method to mitigate bias in encoder-only language models using prompt-tuning, which we evaluate for the problem of gender bias.
- We design and motivate a novel loss function based on the Kullback–Leibler (KL) divergence, which we use for tuning the prompts.
- We provide an extensible set of templates that can be used as a starting point for removing other biases.

2 Related Work

2.1 Bias Quantification Benchmarks

Bias quantification benchmarks aim to measure the bias present in a model towards certain demographics. In LLMs, bias can be quantified using embedding-based metrics, probability-based metrics or generated text metrics.

Embedding-based metrics, such as Word Embeddings Association Test (**WEAT**) (Caliskan et al., 2017) or Sentence Embedding Association Test (**SEAT**) (May et al., 2019) quantify biases by measuring the association between two groups of bias attributes (e.g. associated with male and female terms) and two groups of target attributes (e.g. associated with family and career). SEAT, used for contextual models like BERT or RoBERTa, creates sentence-level embeddings by filling in templates with terms from the four groups.

Probability-based methods, such as **StereoSet** (Nadeem et al., 2021), quantify bias by measuring how frequently a model chooses a stereotypical word to fill in a masked token. In StereoSet, the

[1] Our implementation is available at https://github.com/ChiscaAndrei/prompting-fairness

Proceedings of the Fourth Workshop on Language Technology for Equality, Diversity, Inclusion, pages 52–62
March 21, 2024 ©2024 Association for Computational Linguistics

Nr.	Template
1	<GenderedWord> is a <Target>.
1	<GenderedWord> works as a <Target>.
2	<GenderedWord> worked as an <Target> for two years.
3	<GenderedWord> is a good <Target>.
4	<GenderedWord> earns <HisOrHer> living as a <Target>.
5	I'm glad that <GenderedWord> is a <Target>.
6	<GenderedWord> is studying to be an <Target>.
7	<GenderedWord> had this idea ever since <GenderedWord> was hired as a <Target>."
8	It was hard for <HimOrHer> to become a <Target>.
9	<HisOrHer> career as a <Target> is lucrative.
10	<HisOrHer> job as a <Target> is exhausting.

Table 1: Some examples of templates used for reducing gender bias. Slot names are enclosed by angle brackets.

model can choose from a stereotypical, an anti-stereotypical and an unrelated choice for each scenario. The stereotype score represents the percentage of scenarios where a model prefers the answer that confirms a stereotype.

2.2 Bias Mitigation

Bias mitigation methods aim to reduce the bias in the output of models. Mitigation can occur at different stages during the training or inference or as a separate pre-processing or post-processing step. Attacking the root cause of the biases present in LLMs is often challenging, so most mitigation methods in this context occur after the pretraining.

Pre-processing methods often involve altering existing data via either augmentation, generation or filtering. **Counter-factual Data Augmentation** (**CDA**) (Zmigrod et al., 2019) generates new data samples by swapping the bias-driving terms in existing data. For instance, to reduce gender bias, gender-specific terms (he/she, his/hers) are swapped, and the model undergoes additional pretraining using the new. A visible disadvantage of this method is that it requires updating all the model weights, which might not be trivial for very large models.

Projection-based methods such as **Iterative Nullspace Projection** (**INLP**) (Ravfogel et al., 2020) and **Sentence Debias** (Liang et al., 2020) rely on embedding projection to alter the representation of the input data. Although these two methods do not require additional training, they also have drawbacks. INLP negatively impacts the language modeling ability (Meade et al., 2022), while Sentence Debias requires additional data augmentation.

In-training methods such as architecture modifications (e.g. with adapters - ADELE (Lauscher et al., 2021)), equalizing loss terms (e.g. embedding balancing (Liu et al., 2020)) or additional regularization (e.g. **Dropout** (Webster et al., 2020)) alter the training process of language models. For mitigating biases using Dropout, the model undergoes another round of pretraining with an increased dropout for the attention weights.

3 Prompt Tuning for Bias Mitigation

It has been shown that concatenating prompts to the input of a pretrained language model is a viable method of altering its behaviour for different use cases. Notably, in-context learning, which involves prompting with a few training examples, can be successfully used for adapting a model to various downstream tasks. In (Xie et al., 2022), the authors formalize in-context learning as an *implicit Bayesian inference*, such that the probability of the model's output O can be expressed as

$$p(O|P) = \int_C p(O|C,P)p(C|P)d(C)$$

where the model implicitly infers a *latent concept* C based on the given prompt P.

We argue that a similar approach can be used for debiasing encoder-only LLMs. During pretraining, the model learns to maximize the likelihood of the training data. This behaviour might not always be desirable, especially if, due to the characteristics of the training data, maximizing its likelihood involves relying on various stereotypes. As opposed to removing or hiding information from the model, either at training or at inference time, we aim to give the model additional information at inference, in the form of compact prompt embeddings, which could enable it to implicitly infer a *latent concept* encompassing the desired behaviour: generating a fair and unbiased output.

The prompts should be able to encompass the desired behaviour as accurately as possible. Ideally, we want the model to produce output which is *unbiased* while also retaining the identity of all social groups and maintaining correctness in general language modeling. Trying to express this in handcrafted prompts would not be straightforward, especially if the model to be unbiased was not explicitly trained to follow human instructions. Instead, we base our approach on *"prompt tuning"* (Lester et al., 2021), which involves concatenating a set of trainable embeddings to the embedded input of the model while keeping the other parameters frozen.

Templates The prompt embeddings are trained using a dataset of templates with *bias slots* and *target slots*. Each *bias slot* can be replaced by words specific to each social group affected by the type of bias to be mitigated. For example, for gender bias, we could have a *bias slot* which can be replaced by either "he" or "she", another one which could be replaced by either "his" or "her" and so on. The *target slots* represent the words in the templates which the model should predict. For target slots, there is a set of *allowed options*, composed of:

- *general options* – a set of possible completions for the slot which are the same for each of the social groups considered
- *group specific options* – a set of completions which have a different variant for each of the social groups considered

The reason for explicitly defining the expected outputs of the model and dividing it into *general* and *specific* is to avoid training prompts which cause the model to *"forget"* the identity of each group. For simplicity, we use a single target slot per template.

Training and Loss function We train the prompts by replacing the *bias slots* of each template with their specific variants for each group and minimizing the KL divergence between the probability distribution predicted by the model for the *allowed options* of the *target slots* and a reference probability distribution. For a given template T and social group A, the *bias slots* in T are replaced with corresponding substitutions for A to obtain T_A. We denote by

$$Options_G = \{g_1, g_2, \dots, g_{N_G}\}$$

the set of *general options* and by

$$Options_S(A) = \{s_{A,1}, s_{A,2}, \dots, s_{A,N_S}\}$$

the set of *group-specific options* for group A. Then, denoting by t the *target slot* for T_A, we obtain the probability distribution P_{T_A} defined on the sample space:

$$\begin{aligned}\Omega_{T_A} = \{&t = g_1, \dots, t = g_{N_G},\\ &t = s_{A,1}, \dots, t = s_{A,N_S},\\ &t \notin Options_G \cup Options_S(A)\} \qquad (1)\end{aligned}$$

Here $P_{T_A}(t = x)$ represents the probability predicted by the model for word x in the *target slot* t of T_A. To obtain a proper probability distribution, we also consider the probability of t not being in the set of *allowed options*.

We choose the *reference probability distribution* $P^*_{T_A}$ for T_A as the average probabilities predicted by the original model (denoted by P^i) across the set $\boldsymbol{G}$ of all social groups considered:

$$P^*_{T_A}(t = g_k) = \frac{1}{|\boldsymbol{G}|} \sum_{G \in \boldsymbol{G}} P^i_{T_G}(t = g_k) \qquad (2)$$

$$P^*_{T_A}(t = s_{A,k}) = \frac{1}{|\boldsymbol{G}|} \sum_{G \in \boldsymbol{G}} P^i_{T_G}(t = s_{G,k}) \qquad (3)$$

We define the loss term for template instantiation T_A, obtained by filling *bias slots* in template T with terms specific for social group G, as the KL divergence between the probability distribution predicted by the model and the *reference probability distribution*:

$$L_{T_A} = D_{KL}(P_{T_A} \parallel P^*_{T_A}) \qquad (4)$$

The *reference probability distribution* $P^*_{T_A}$ for each template instantiation T_A is treated as a constant and can be precomputed beforehand.

Then, the total loss for a set $\boldsymbol{T}$ of templates is obtained by instantiating each template T with the *bias slot terms* for each social group G, and summing over the loss terms for each resulting template instantiation T_G:

$$L = \sum_{T \in \boldsymbol{T}} \sum_{G \in \boldsymbol{G}} L_{T_G} \qquad (5)$$

BiasSlotName	Male Variant	Female Variant
GenderedWord	he	she
	Robert	Patricia
	Michael	Jennifer
	William	Barbara
	Richard	Susan
	Daniel	Jessica
	Andrew	Karen
	George	Emily
	Brian	Rebecca
	Ryan	Cynthia
	Stephen	Emma
HeOrShe	he	she
HisOrHer	his	her
HimOrHer	him	her

Table 2: Gender *bias slots* used in templates

In previous formulas, we assumed for simplicity a single template instantiation T_G for each pair of a template T and a group G. In the general case, there may exist multiple such template instantiations T_G^k, depending on whether some *bias slots* in T can be filled by multiple pairs of values. In this case, we sum over all considered[2] instantiations:

$$L = \sum_{T \in \boldsymbol{T}} \sum_{G \in \boldsymbol{G}} \sum_{k} L_{T_G^k} \tag{6}$$

Gender debiasing BERT and RoBERTa We particularize this method for reducing gender bias in BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019) models. We constructed a dataset of 159 templates, mostly focused on genders in relation to professions/occupations, as this is one area in which we empirically observed the models to generate biased predictions. Examples of templates are listed in Table 1. We use 4 types of bias slots, as described in Table 2.

For simplicity, we restrict the choice of *allowed options* to words that each model's tokenizer can represent with a single token and subsequently replace the *target slots* in the templates by a single [MASK] token. This is not a big limitation in this case, since BERT's and RoBERTa's vocabularies can represent most common English words by a single token. However, it might pose problems for models with other types of tokenizers and for different languages, as it would require either using a limited number of options or creating separate templates for different numbers of mask tokens. We selected the *allowed options* empirically by hand-picking appropriate completions and choosing from the original model's predictions for the templates. While templates are focused on professions/occupations, the *allowed options* are not restricted only to this specific domain; for some of the templates, there are other valid completions. We selected 219 *general options* and 15 pairs of *group specific options*. Some examples are listed in Table 11 and Table 12.

Implementation We use pretrained models from Huggingface Transformers (Wolf et al., 2020) and use the *prompt tuning* implementation in from PEFT (Mangrulkar et al., 2022) with PyTorch (Paszke et al., 2019) for building and training our debiased models.

Training is done using an AdamW (Loshchilov and Hutter, 2019) optimizer with an initial learning rate of $1e-2$ and a linearly decreasing schedule with warmup. Since only prompt parameters are updated and the dataset used is small, training converges fairly rapidly: training of each debiased model takes about half an hour on an Nvidia 1050Ti GPU.

4 Results

We evaluate our method for mitigating gender bias in BERT and RoBERTa on the gender tests from SEAT (May et al., 2019) and StereoSet (Nadeem et al., 2021). For StereoSet, a *stereotype score (SS)* closer to 50% indicates a less biased model. In case of SEAT, we average the last layer's hidden representations and normalize the resulting vector, as May et al. (2019); Meade et al. (2022), but *exclude* the representations corresponding to the prompt tokens from this computation. For analyzing the loss in language modeling performance, we use the *language modeling (LM)* score from StereoSet and the pseudo-perplexity (Salazar et al., 2020) on the *test* split of WikiText-2 (Merity et al., 2017). For computing the pseudo-perplexity, we first sentencize each text in the dataset, using Spacy (Honnibal et al., 2020).

Initialization method In preliminary experiments with BERT, using *random initialization* for the prompt's parameters, we observed, similarly to Lester et al. (2021), that the prompts learned and the performance depend to a large extent on the initialization. We also examined for each

[2] for practical reasons, we might consider only a subset of all possible instantiations

	SEAT Gender Avg. Effect Size (↓)	**StereoSet Gender SS Score** (%)	**StereoSet LM Score** (↑)
BERT base uncased	0.620	60.279	84.172
+ PF Random init.	0.393 ±0.068(0.095)	58.901 ±0.437(0.597)	84.036 ±0.146(0.204)
+ PF Gendered init.	0.455 ±0.118(0.095)	58.605 ±0.495(0.399)	**84.576** ±0.226(0.182)
+ PF Neutral init.	0.454 ±0.092(0.074)	58.675 ±0.349(0.281)	84.333 ±0.273(0.220)
+ PF FemaleBiased init.	**0.330** ±0.071(0.057)	**58.456** ±0.674(0.543)	84.460 ±0.155(0.125)
+ CDA	0.722	59.610	83.080
+ Dropout	0.765	60.660	83.040
+ INLP	**0.204**	**57.250**	80.630
+ SentenceDebias	0.434	59.370	84.200
+ Self-Debias	-	59.340	84.090
RoBERTa base	0.940	66.323	88.929
+ PF Random init.	0.838 ±0.042(0.059)	65.495 ±0.677(0.946)	88.729 ±0.121(0.169)
+ PF Gendered init.	0.686 ±0.075(0.060)	64.186 ±1.018(0.820)	**89.008** ±0.284(0.229)
+ PF Neutral init.	**0.635** ±0.067(0.054)	**63.939** ±0.614(0.495)	88.908 ±0.300(0.241)
+ PF FemaleBiased init.	0.702 ±0.040(0.032)	64.319 ±0.529(0.426)	88.944 ±0.150(0.121)
+ CDA	0.880	64.430	88.830
+ Dropout	1.074	66.260	88.810
+ INLP	0.823	**60.820**	88.230
+ SentenceDebias	0.846	62.770	88.940
+ Self-Debias	-	65.040	88.260

Table 3: Results of gender debiased models with different initialization types compared with results reported by Meade et al. (2022) for CDA, Dropout, INLP, SentenceDebias and Self-Debias. Our results are averaged across all trials, with a 95% *confidence interval (±)* and with the standard deviation in parentheses. For SEAT, we report the *mean absolute effect sizes* across all 6 gender tests. For StereoSet, we report the *Stereotype Score (SS)* for *gender* test and *Language Modeling Score (LM)* across all tests.

prompt token the closest[3] 5-word embeddings in the model's vocabulary, before and after training, and remarked that in some cases, the model[4] tends to learn prompts close to *female gendered* words.

Based on these preliminary findings, we evaluated the performance of 4 different types of initialization methods, using a prompt length of 3 tokens in each case:

- Random initialization – prompt embeddings are initialized randomly[5].

- Neutral initialization – each prompt token's embedding is initialized with a *neutral* world, unrelated to genders.

- Gender Balanced initialization – one prompt token's embedding is initialized with the embedding of a word related to the *male gender*, one is initialized to the embedding of a *neutral* word, and one is initialized with the embedding of a word related to the *female gender*.

- Female Biased initialization – each prompt token's embedding is initialized with the embedding of a word related to the *female gender*.

Words used for each type of initialization are listed in Tables 9,10.

For this experiment, we use *bert-base-uncased* and *roberta-base* as base models and don't use any names in the *bias slots* of the templates (the <GenderedWord> slots are filled only with "he" or "she"). Given each base model, we train 10 models using *Random*, 5 with *Neutral* initialization, 5 with *Gender Balanced* initialization and 5 with *female Biased* initialization. Each model is trained for 250 epochs, with batches of 16 templates. In Table 3, we report the mean and standard deviation across each type of initialization and compare the results with those reported by Meade et al. (2022) for gender debiasing using CDA, Dropout, INLP,SentenceDebias and Self-

[3] in terms of *cosine distance*

[4] preliminary experiments were only performed for BERT

[5] using default Embedding initialization in PyTorch: normal distribution with mean 0 and standard deviation 1

	Profession SS (%)	**Race SS** (%)	**Religion SS** (%)
BERT base uncased	58.934	57.030	59.704
+ PF Random init.	57.227 ±0.172(0.240)	56.978 ±0.200(0.279)	60.437 ±0.590(0.825)
+ PF Gendered init.	56.942 ±0.294(0.237)	**56.595** ±0.227(0.183)	59.755 ±0.946(0.762)
+ PF Neutral init.	**56.940** ±0.246(0.198)	56.833 ±0.887(0.714)	**59.358** ±1.468(1.182)
+ PF FemaleBiased init.	57.026 ±0.336(0.271)	56.842 ±0.272(0.219)	59.362 ±0.620(0.499)
RoBERTa base	61.467	61.674	64.278
+ PF Random init.	60.893 ±0.306(0.427)	61.773 ±0.213(0.298)	64.432 ±0.769(1.074)
+ PF Gendered init.	59.736 ±0.580(0.467)	61.591 ±0.249(0.200)	62.870 ±1.327(1.068)
+ PF Neutral init.	**59.663** ±0.352(0.283)	61.390 ±0.861(0.694)	**61.804** ±1.515(1.220)
+ PF FemaleBiased init.	59.680 ±1.109(0.893)	**61.324** ±0.633(0.509)	63.391 ±1.275(1.027)

Table 4: Stereotype scores of *gender debiased* models with different initialization types on StereoSet *profession, race* and *religion* tests. Results are averaged across all trials, with a $95\%confidenceinterval$ (±) and the standard deviation (in parentheses).

	SEAT Gender Avg. Effect Size (↓)	**StereoSet Gender SS Score** (%)	**StereoSet LM Score** (↑)	**Pseudo-Perplexity** (↓)
BERT base uncased	0.620	60.279	84.172	6.396
+ With Names	0.397 (0.060)	58.854 (0.804)	84.347 (0.269)	**6.517** (0.044)
+ Without Names	**0.330** (0.057)	**58.456** (0.543)	**84.460** (0.125)	6.530 (0.088)
BERT base cased	0.686	61.229	82.522	**5.542**
+ With Names	**0.414** (0.086)	59.163 (0.403)	82.494 (0.108)	5.786 (0.140)
+ Without Names	0.587 (0.131)	**59.123** (0.452)	**82.422** (0.170)	5.788 (0.145)

Table 5: Results of gender debiased models with and witthe hout usage of names in the training dataset, for two types of base models: bert-base-uncased and bert-base-cased. *Female biased* initialization is used in all cases. For training *with names*, models are trained for 40 epochs and 250 otherwise. Results are averaged over all 5 different initializations, with standard deviation in parentheses.

	PseudoPerplexity (↓)
BERT base uncased	6.396
+ PF Random init.	6.584 ±0.117(0.164)
+ PF Gendered init.	6.674 ±0.272(0.219)
+ PF Neutral init.	6.572 ±0.128(0.103)
+ PF FemaleBiased init.	**6.530** ±0.109(0.088)
RoBERTa base	11.198
+ PF Random init.	11.464 ±0.154(0.215)
+ PF Gendered init.	**11.146** ±0.398(0.320)
+ PF Neutral init.	11.410 ±0.153(0.123)
+ PF FemaleBiased init.	11.472 ±0.571(0.460)

Table 6: Pseudo-perplexities of models gender debiased using different methods of initialization. Pseudo-perplexities are computed on the *test* split of WikiText-2. We sentencize each text before computing the pseudo-perplexities. Results are averaged across all trials, with a $95\%confidenceinterval$ *(±)* and standard deviation (in parentheses).

DEBIAS. The pseudo-perplexities are listed in Table 6.

In case of **BERT**, we remark that among the different initialization methods, the *female biased* initialization yields the best results both in terms of *debiasing* and retaining of language modeling performance. Between *random*, *gendered balanced* and *neutral* initialization, the results are similar overall. Compared to other debiasing techniques, the *female biased* initialization is second to INLP in terms of debiasing, while the other initialization types are on par with SENTENCEDEBIAS. However, our method generally results in a good *language modeling* score and limited increase in pseudo-perplexity, while INLP significantly reduces the LM score. We suspect this might be due to INLP removing all gender-related information for the model's output.

In case of **RoBERTa**, we notice that the *neutral* initialization achieves better results than the other initialization types. While this achieves significantly better results on SEAT compared to all other debiasing techniques, its results on StereoSet are surpassed by both INLP and SENTENCEDEBIAS. As for BERT, results show that our method generally succeeds in maintaining the language modelling ability of the base model.

Effect on other types of biases Besides mitigating the targeted bias and the impact on language modeling performance, the side effects on other biases should also be considered. We evaluate our *gender debiased* models on the *profession*, *race* and *religion* tests in StereoSet and report the *stereotype scores* in Table 4.

In the *profession bias*, test there is a significant decrease in bias for all initialization methods, most probably due to the dataset used for training, which is focused on genders and professions. There is no significant effect for *race bias*. For *religion bias*, the results vary notably across different trials with the same type of initialization, and there is, on average, an increase in bias for models trained using *random initialization*. These results suggest extending this method for targeting multiple biases might be feasible.

Using names in training In previous experiments, all *bias slots* in templates were replaced with gendered pronouns. Besides pronouns, more types of words contain gender-related information, such as names and gendered nouns. We experiment with adding names to the training dataset by also replacing <GenderedWord> bias slots with names. For each template containing a <GenderWord> slot, we instantiate it once with the *("he", "she")* pair and 10 more times with pairs of one male name and one female name. While the names are the same, their pairing is different for each template. Since names in English are capitalized, we evaluate our results both for *bert-base-uncased* and *bert-base-cased*, as we presume the capitalization might give the models a better understanding for the concept of 'names'. We only evaluated *female biased* initialization because it achieved better results in previous experiments. To account for different dataset sizes, we train for 40 epochs when using names and for 250 epochs otherwise. Results are listed in Table 5.

For the *uncased* model, debiasing results on SEAT and StereoSet are marginally better without using names, while the language modeling performance is roughly the same. In the case of the *cased* model, we notice that using names yields a slightly better performance on SEAT, while the stereotype scores for StereoSet and the language modeling ability remain roughly the same. This might be because half of the SEAT tests use names as gender attributes.

Ablation for *group specific options* We investigate the effect *gender specific options* have on debiasing and language modeling. We evaluate our *gender debiased* models based on *bert-base-uncased*, using *female biased* initialization, with and without using *group specific options*. Results are presented in Table 7. For SEAT, we observe similar results in both cases, while in the case of StereoSet, the results *without* group-specific options are significantly better. However, using *group specific options* results in a better language modeling performance that can be observed through the pseudo-perplexity and StereoSet LM score.

In addition to these benchmarks, we analyzed the predictions of the debiased models on our training dataset and noticed that when trained *without gender-specific options*, the models tend to "forget" about gender information, assigning female-gendered words to male template instantiations and vice-versa. Some examples are presented in Table 8. Even though a model that *forgets* the identity of social classes might be considered *unbiased*, we argue that such behaviour would be undesirable for many applications.

5 Conclusions

We proposed and investigated a method of reducing social biases in pretrained LLMs based on prompt tuning, which involves training the prompt embeddings on a small set of templates. In addition to debiasing, this method is also designed to prevent the model from 'forgetting' the identity of the social groups targeted during debiasing.

We experimented with this method for reducing *gender bias* in BERT and RoBERTa models. Results show that it achieves debiasing performance on par with state-of-the-art while maintaining information about the social group's identity and with a low impact on general language modeling performance. While we focused our experiments on mitigating a single type of social bias in BERT and RoBERTa, the method can be adapted to other types of biases and LLMs.

	SEAT Gender Avg. Effect Size (↓)	**StereoSet Gender SS Score** (%)	**StereoSet LM Score** (↑)	**Pseudo-Perplexity** (↓)
BERT base uncased	0.620	60.279	84.172	**6.396**
+ Only General opts.	0.335 (0.086)	**57.600** (0.549)	84.174 (0.235)	6.746 (0.271)
+ Group specific opts.	**0.330** (0.057)	58.456 (0.543)	**84.460** (0.125)	6.530 (0.088)

Table 7: Results of gender debiased models with and without using *group specific options* in training. *Female biased* initialization is used. Results are averaged over all 5 different initializations, with standard deviation in parentheses.

		Without GSO		With GSO	
Nr.	**Template**	**Male**	**Female**	**Male**	**Female**
1	<GenderedWord> was an efficient <Target>.	woman	woman	man	woman
2	<GenderedWord> was an unskilled <Target>.	woman	woman	man	woman
3	<GenderedWord> was a good <Target>.	girl	girl	man	woman
4	<GenderedWord> is no longer a lazy <Target>.	man	woman	man	woman
5	<GenderedWord> had this idea ever since <HeOrShe> was hired as a <Target>.	waitress	nurse	nurse	waitress
6	<HisOrHer> career as an <Target> is boring.	actress	actor	actor	actress
7	<HisOrHer> career as an <Target> is well paid.	actor	actor	actor	actress
8	<HisOrHer> career as an <Target> is unappreciated.	actress	actress	actor	actress
9	<HisOrHer> job as a <Target> is poorly paid.	waitress	waitress	waiter	waitress

Table 8: Examples of predictions from a bert-base-uncased model *gender debiased* with *female biased* initialization, both *with* and *without* group specific options (GSO). For each template, we show the top model prediction of each debiased model variant on the *female* and *male* template instantiations.

Limitations

When implementing our method for BERT and RoBERTa, we considered, for simplicity, a set of *allowed options*, which can be represented by a single token in these model's vocabulary. This is not too restrictive in our case since their tokenizers can represent the most common English words with a single token. However, it might prove limiting for debiasing models with other types of tokenizers and usage in other languages. We note that while the general concept of our method could be applied even for *allowed options* that can span multiple tokens, such an implementation is not straightforward for all models. Proper computation of all probabilities used in the loss function might require a separate pass through the model for each *allowed option*. Future investigation is required to determine the feasibility of our method in such cases and to design efficient and numerically stable implementations.

In this paper, we focused our experiments only on mitigating gender bias. Our theoretical approach can be utilized for other types of social biases, but doing so in practice would require creating *templates* and selecting appropriate *general* and *group-specific options* for each type of bias targeted. We note that for some types of biases, this might not be straightforward, especially if the number of social groups considered is large, and we deem it probable for the overall performance of the method to be limited by the quality of the dataset used. A possible future improvement could be to find a method of automatically extracting relevant *templates* and *allowed options* from existing large datasets.

Experiments have shown that the performance of models debiased using our approach depends to a large extent on the used initialization method. Results show that for *gender debiasing* BERT, initializing with terms related to the *female* gender gives better results on average than random initialization and other approaches, while in the case of RoBERTa the *neutral* initialization achieves the best results. However, other initialisation methods might be more suitable, and this approach is not directly usable for other biases. Further investigation into robust initialization methods is needed.

The loss function of our method considers the *reference probability distribution* as the average of distributions predicted by the original model for each social group considered. While this approach is reasonable, it might prove limiting in some cases. For example, an exceedingly biased or toxic model could predict unfair probability distributions for some social groups, which would skew the average.

References

Aylin Caliskan, Joanna J. Bryson, and Arvind Narayanan. 2017. Semantics derived automatically from language corpora contain human-like biases. *Science*, 356(6334):183–186.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

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

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, Adriane Boyd, et al. 2020. spacy: Industrial-strength natural language processing in python.

Anne Lauscher, Tobias Lüken, and Goran Glavaš. 2021. Sustainable modular debiasing of language models.

Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3045–3059, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

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

Haochen Liu, Jamell Dacon, Wenqi Fan, Hui Liu, Zitao Liu, and Jiliang Tang. 2020. Does gender matter? towards fairness in dialogue systems.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. Cite arxiv:1907.11692.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Sourab Mangrulkar, Sylvain Gugger, Lysandre Debut, Younes Belkada, and Sayak Paul. 2022. Peft: State-of-the-art parameter-efficient fine-tuning methods. `https://github.com/huggingface/peft`.

Chandler May, Alex Wang, Shikha Bordia, Samuel R. Bowman, and Rachel Rudinger. 2019. On measuring social biases in sentence encoders. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 622–628, Minneapolis, Minnesota. Association for Computational Linguistics.

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

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. 2017. Pointer sentinel mixture models. In *International Conference on Learning Representations*.

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

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32.

Shauli Ravfogel, Yanai Elazar, Hila Gonen, Michael Twiton, and Yoav Goldberg. 2020. Null it out: Guarding protected attributes by iterative nullspace projection. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 7237–7256. Association for Computational Linguistics.

Julian Salazar, Davis Liang, Toan Q. Nguyen, and Katrin Kirchhoff. 2020. Masked language model scoring. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2699–2712, Online. Association for Computational Linguistics.

Kellie Webster, Xuezhi Wang, Ian Tenney, Alex Beutel, Emily Pitler, Ellie Pavlick, Jilin Chen, Ed H. Chi, and Slav Petrov. 2020. Measuring and reducing gendered correlations in pre-trained models. Technical report.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Sang Michael Xie, Aditi Raghunathan, Percy Liang, and Tengyu Ma. 2022. An explanation of in-context learning as implicit bayesian inference. In *International Conference on Learning Representations*.

Ran Zmigrod, Sabrina J. Mielke, Hanna Wallach, and Ryan Cotterell. 2019. Counterfactual data augmentation for mitigating gender stereotypes in languages with rich morphology. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1651–1661, Florence, Italy. Association for Computational Linguistics.

A Implementation details

In this appendix, we present some additional details related to the implementation of our method for gender debiasing BERT and RoBERTa.

As described in Section 4, we experimented with several types of initialization for the prompt: *random*, *neutral*, gender balanced and *female biased*. The words used as initialization were chosen such that they can be represented as a single token. Due to this, they differ slightly between the two models. These are shown in Table 9 (for BERT) and in Table 10 (for RoBERTa).

Training requires the selection of a set of *allowed options* (composed of *general options* and *group specific options* for the *target slots*. We selected these manually, mostly by choosing from the most likely predictions of the *BERT base* model on our set of templates. Some examples are shown in Table 11 and Table 12.

	Neutral	
tree	stone	lake
animal	mountain	house
fair	water	balanced
justice	equality	human
honest	forest	gold
	Gender Balanced	
man	water	woman
girl	fair	boy
masculine	human	feminine
female	justice	male
husband	big	queen
	Female Biased	
women	queen	girl
female	priestess	feminine
wife	feminist	mothers
feminism	miss	suffrage
she	women	mothers

Table 9: Words used as initialization for **BERT** prompts in each trial, for our chosen test initialization types

.

	Neutral	
tree	stone	lake
animal	mountain	house
fair	water	balanced
justice	equality	human
honest	forest	gold
	Gender Balanced	
man	water	woman
girl	fair	boy
masculine	human	feminine
female	justice	male
husband	big	queen
	Female Biased	
women	queen	girl
female	her	feminine
wife	feminist	mothers
woamn	miss	female
she	women	mothers

Table 10: Words used as initialization for **RoBERTa** prompts in each trial, for our chosen test initialization types

.

Nr	**Option**
1	teacher
2	doctor
3	engineer
4	entrepreneur
5	person
6	scientist

Table 11: Examples of *general options* used for gender debiasing

Nr.	**Male Variant**	**Female Variant**
1	actor	actress
2	waiter	waitress
3	husband	wife
4	boy	girl
5	man	woman

Table 12: Examples of *group specific options* used for gender debiasing