Prompt Tuning Pushes Farther, Contrastive Learning Pulls Closer: A Two-Stage Approach to Mitigate Social Biases

Yingji Li[1], **Mengnan Du**[2], **Xin Wang**[3*], **Ying Wang**[1,4*]

[1]College of Computer Science and Technology, Jilin University, Changchun, China
[2]Department of Data Science, New Jersey Institute of Technology, Newark, USA
[3]School of Artificial Intelligence, Jilin University, Changchun, China
[4]Key Laboratory of Symbolic Computation and Knowledge Engineering
of Ministry of Education, Jilin University, Changchun, China

yingji21@mails.jlu.edu.cn, mengnan.du@njit.edu, {xinwang,wangying2010}@jlu.edu.cn

Abstract

As the representation capability of Pre-trained Language Models (PLMs) improve, there is growing concern that they will inherit social biases from unprocessed corpora. Most previous debiasing techniques used Counterfactual Data Augmentation (CDA) to balance the training corpus. However, CDA slightly modifies the original corpus, limiting the representation distance between different demographic groups to a narrow range. As a result, the debiasing model easily fits the differences between counterfactual pairs, which affects its debiasing performance with limited text resources. In this paper, we propose an adversarial training-inspired two-stage debiasing model using Contrastive learning with Continuous Prompt Augmentation (named CCPA) to mitigate social biases in PLMs' encoding. In the first stage, we propose a data augmentation method based on continuous prompt tuning to push farther the representation distance between sample pairs along different demographic groups. In the second stage, we utilize contrastive learning to pull closer the representation distance between the augmented sample pairs and then fine-tune PLMs' parameters to get debiased encoding. Our approach guides the model to achieve stronger debiasing performance by adding difficulty to the training process. Extensive experiments show that CCPA outperforms baselines in terms of debiasing performance. Meanwhile, experimental results on the GLUE benchmark show that CCPA retains the language modeling capability of PLMs.

1 Introduction

Pre-trained Language Models (PLMs) have demonstrated outstanding performance in recent years and have been widely used in natural language understanding tasks (Peters et al., 2018; Delobelle et al., 2022). However, the powerful language modeling capability enables PLMs to learn good representations from large-scale training corpora while capturing human-like social biases. Recent studies have demonstrated that the representations encoded by PLMs learn social biases specific to demographic groups (e.g., gender, race, religion) and can be amplified to downstream tasks, leading to unfair outcomes and adverse social effects (Zhao et al., 2019; Webster et al., 2020). As a result, mitigating social biases in PLMs' encoding can improve the fairness of NLP systems significantly (Bolukbasi et al., 2016; Bender and Friedman, 2018).

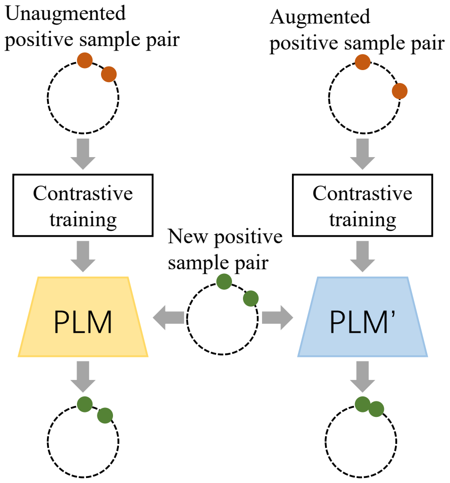

Figure 1: The motivation of CCPA. For the new input samples, the PLM's performance of the augmented sample training is stronger than that of the unaugmented sample training.

Most existing debiasing techniques first need to construct sample pairs using Counterfactual Data Augmentation (CDA) (Zmigrod et al., 2019; Wang et al., 2022) to balance the training corpora. The general approach of CDA is to replace the original corpus with attribute words (e.g., *he*/*she*, *man*/*woman*) specific to different demographic groups. For example, RCDA (Chen et al., 2021) uses a generator to generate a large number of antisense sentences and then uses a discriminator

[*]Corresponding author

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 14254–14267
July 9-14, 2023 ©2023 Association for Computational Linguistics

to evaluate the quality of the original and antisense samples jointly. FairFil (Cheng et al., 2021) obtains a pair of positive sample sentences by replacing the attribute words in the training corpora with the antonyms and then uses contrastive learning to train a filter for debiasing. Auto-Debias (Guo et al., 2022) uses pairs of attribute words as training corpora, amplifies the bias between sample pairs by searching biased prompt texts in the Wikipedia vocabulary, and then performs semantic alignment using Jensen-Shannon divergence. These methods aim to mitigate social biases between different demographic groups by narrowing the representation distance between sample pairs. However, CDA slightly modifies the original corpus, limiting the representation distance between different demographic groups to a narrow range. As a result, the debiasing model is easy to overfit the difference between counterfactual pairs, which affects its learning ability with limited text resources. As shown in Figure 1, it is difficult for PLMs to achieve the ideal debiasing performance for newly input samples with greater difficulty.

In this work, we propose a two-stage debiasing method using Contrastive learning with Continuous Prompt Augmentation (named CCPA) to mitigate social biases in PLMs' encoding. Inspired by adversarial training, our approach improves the debiasing ability of PLMs by first amplifying and then attenuating the bias between different demographic groups. Specifically, we first use CDA to replace attribute words in the original training corpus to construct counterfactual pairs corresponding to different demographic groups. In the first stage, we augment the positive sample pairs with continuous prompt tuning to increase the distance between them to amplify the biases between different demographic groups. In the second stage, we utilize contrastive learning to pull the distance between the positive sample pairs to attenuate the biases between different demographic groups. CCPA increases the difficulty of model fitting by expanding the representation space between sample pairs. We believe that difficult learning experiences make the model more powerful, thus improving the debiasing ability of PLMs training in corpora with limited resources. Our main contributions are as follows:

- We propose the CCPA debiasing framework that combines prompt tuning and contrastive learning to learn a debiased PLM representation. The PLM's parameters are fixed in the first stage, and a generator encoding continuous prompts is trained. In the second stage, the prompts are fixed, and the PLM's parameters are fine-tuned using contrastive learning.
- We propose data augmentation using continuous prompts to achieve excellent debiasing performance using small training data rather than relying on a large external corpus. Given that continuous prompts may cause the representation distance between sample pairs to be too far apart, causing the semantic space to degrade, we propose constraining the prompt tuning using the Mahalanobis Distance to keep the semantic space as stable as possible.
- We train CCPA on several real-world corpora and mitigate bias on the most common gender bias. The results on BERT and DistilBERT show that CCPA is superior to state-of-the-art models. In addition, we test the downstream tasks on the GLUE benchmark, and show that CCPA retains the language modeling capability while improving the PLMs' fairness.

2 Methodology

In this section, we propose the Contrastive learning with Continuous Prompt Augmentation (CCPA) framework to mitigate the social bias in the encoding of PLMs specific to the most common gender bias. Our proposed CCPA consists of two stages: 1) Continuous Prompt Tuning and 2) Fine-Tuning with Contrastive Learning. The framework of CCPA is shown in Figure 2.

2.1 Pre-Processing based on CDA

First, we pre-process the training corpus with imbalanced samples using Counterfactual Data Augmentation (CDA). Given a list of attribute words specific to gender bias,[1] for each attribute word (e.g., *male/female*), we match sentences containing an attribute word in the training corpus. The attribute word is then replaced with the opposite word in a different gender direction (e.g., *male* is replaced by *female*), leaving the other words unchanged. Then, we get the pre-processed training corpus $\mathcal{S} = \{(s_1, s'_1), (s_2, s'_2), \cdots, (s_N, s'_N)\}$ consists of N counterfactual pairs (s_i, s'_i) along different gender directions.

[1] We only consider the binary gender direction and use the same list of gender-specific attribute words as (Bolukbasi et al., 2016; Liang et al., 2020; Cheng et al., 2021).

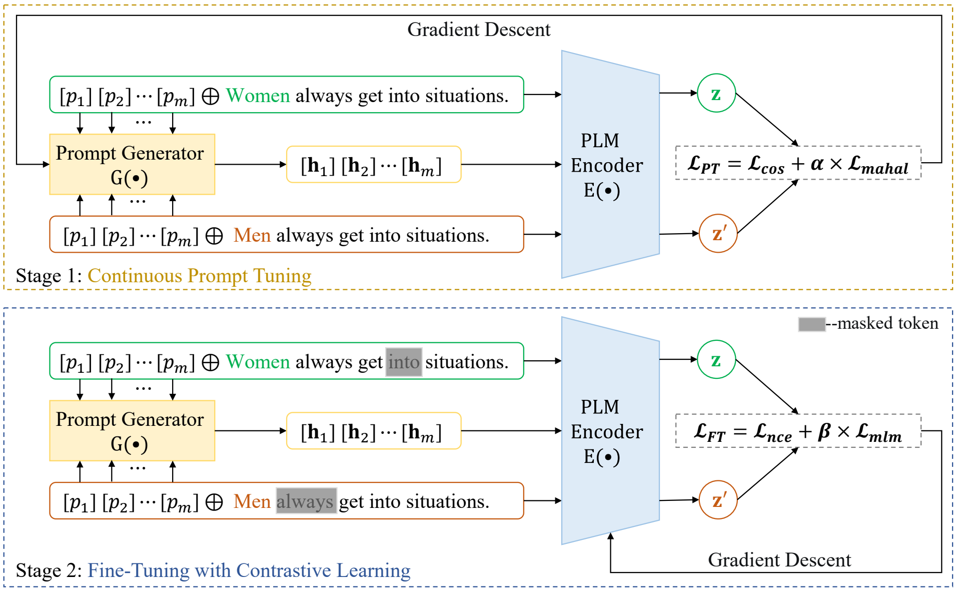
Gradient Descent
$[p_1]\ [p_2] \cdots [p_m] \oplus$ Women always get into situations.
Prompt Generator G(•)
$[\mathbf{h}_1]\ [\mathbf{h}_2] \cdots [\mathbf{h}_m]$
$[p_1]\ [p_2] \cdots [p_m] \oplus$ Men always get into situations.
PLM Encoder E(•)
z
z'
$\mathcal{L}_{PT} = \mathcal{L}_{cos} + \alpha \times \mathcal{L}_{mahal}$
Stage 1: Continuous Prompt Tuning
▆--masked token
$[p_1]\ [p_2] \cdots [p_m] \oplus$ Women always get into situations.
Prompt Generator G(•)
$[\mathbf{h}_1]\ [\mathbf{h}_2] \cdots [\mathbf{h}_m]$
$[p_1]\ [p_2] \cdots [p_m] \oplus$ Men always get into situations.
PLM Encoder E(•)
z
z'
$\mathcal{L}_{FT} = \mathcal{L}_{nce} + \beta \times \mathcal{L}_{mlm}$
Gradient Descent
Stage 2: Fine-Tuning with Contrastive Learning

Figure 2: The overall architecture of CCPA. In the first stage, the parameters of PLM encoder $E(\cdot)$ are fixed and a prompt generator $G(\cdot)$ encoding the continuous prompts is trained, where the goal is to enlarge the bias of sentence pairs. In the second stage, the parameters of the prompt generator $G(\cdot)$ are fixed and the parameters of PLM encoder $E(\cdot)$ are fine-tuned using contrastive loss. Ultimately, we can obtain the debiased PLM encoder $E(\cdot)$.

2.2 Continuous Prompt Tuning

Prompt-based learning is similar to giving instructions to the model task to guide the model learning knowledge more directly (Petroni et al., 2019). A lot of work utilize manually constructed prompts (Schick and Schütze, 2020, 2021) or automatically searched discrete prompts (Shin et al., 2020) to assist language models. However, manually constructed templates are heavily based on the designers' experience and automatically searched prompts are limited by the search space (Liu et al., 2021a). Instead of limiting the prompts to human interpretable natural language, the continuous prompts (Li and Liang, 2021; Zhong et al., 2021) guide directly within the embedding space of the model. Meanwhile, continuous prompts tune their parameters, removing the constraint of templates being parameterized by PLMs' parameters.

Inspired by adversarial training, we believe that increasing the difficulty of the training process can guide the model in acquiring a stronger learning ability. To achieve this goal, we propose a data augmentation method based on continuous prompt tuning to further push the differences between counterfactual pairs. Data augmentation method based on continuous prompt tuning adds difficult information to the model by concatenating embeddings that amplify bias across different demographic groups over counterfactual pairs.

Given a template $T = \{[p_1], [p_2], \cdots, [p_m], s\}$, where s denotes a sentence, $[p_j]$ is a virtual token represented as $[PROMPT]$ and m virtual tokens form a prompt sequence $\mathcal{P}$. For each counterfactual pair $(s_i, s_i') \in \mathcal{S}$ obtained by data preprocessing, we concatenate the same prompt sequence $\mathcal{P}$ at the head of each sentence (see Figure 2). The augmented sample pair is denoted by $(\hat{s}_i, \hat{s}_i{}')$ and is fed into a PLM to obtain the sentence representation. Formally, let M denote a PLM whose encoder $E(\cdot)$ encodes an input sentence $\hat{s}_i$ and outputs a sentence embedding $\mathbf{z}_i = E(\hat{s}_i)$. Similarly, $\mathbf{z}_i' = E(\hat{s}_i{}')$. In order to obtain continuous prompt embeddings, we train a generator $G(\cdot)$ to encode the prompt sequence $\mathcal{P}$. Following P-Tuning (Liu et al., 2021b), we choose a bidirectional long-short-term memory network (LSTM), which consists of a two-layer multilayer perceptron (MLP) and a ReLU activation layer. The embedding $\mathbf{h}_j$ of each virtual token $[p_j]$ in the prompts sequence is encoded by $G(\cdot)$ as follows:

$$\begin{aligned}\mathbf{h}_j &= G([\overrightarrow{\mathbf{h}_j} : \overleftarrow{\mathbf{h}_j}]) \\ &= G([LSTM(\mathbf{h}_{1:j}) : LSTM(\mathbf{h}_{j:m+1})]).\end{aligned} \tag{1}$$

Afterwards, we replace the continuous prompt embeddings $\{\mathbf{h}_1, \mathbf{h}_2, \cdots, \mathbf{h}_m\}$ to the corresponding positions of the sentence embeddings $\mathbf{z}_i$ to obtain the sentence representations pairs $(\mathbf{z}_i, \mathbf{z}_i')$.

In this stage, our training objective is to push away the distance of representation $(\mathbf{z}_i, \mathbf{z}'_i)$ between sample pairs $(\hat{s_i}, \hat{s_i}')$. Briefly, we take the Cosine Similarity between sentence representations as the loss function, defined as follows:

$$\mathcal{L}_{cos} = \frac{\mathbf{z} \cdot \mathbf{z}'}{\|\mathbf{z}\|\|\mathbf{z}'\|} = \frac{\sum_{i=1}^{n} \mathbf{z}_i \cdot \mathbf{z}'_i}{\sqrt{\sum_{i=1}^{n} \mathbf{z}_i^2}\sqrt{\sum_{i=1}^{n} \mathbf{z}_i'^2}}, \quad (2)$$

where $\mathbf{z}$ and $\mathbf{z}'$ denote sentence representations with different sensitive attributes within a batch of size n, respectively. The representation distance between the sample pairs is enlarged with the gradient of similarity decreasing, thus amplifying the bias information between different genders.

Considering that the sentence representation with high-dimensional linear distribution is not independently and equally distributed among the dimensions, only relying on Euclidean distance training may cause the sentence representation to deviate from the original distribution and thus destroy the semantic information. To constrain the change of sentence representation within the original distribution, Mahalanobis distance is taken as the regularization term of the loss function:

$$\mathcal{L}_{mahal} = \sqrt{(\mathbf{z} - \mathbf{S})^{\top} \Sigma^{-1} (\mathbf{z} - \mathbf{S})}, \quad (3)$$

where $\mathbf{z}$ is the representation of a batch size of samples with concatenated prompt embeddings, $\mathbf{S}$ is the representation of the entire pre-processed training samples without concatenated prompt embeddings, and Σ is the covariance matrix of $\mathbf{S}$. Mahalanobis distance is a correction of the Euclidean distance, which corrects the assumption that the Euclidean distance is independent and equally distributed among all dimensions. With the constraint of Mahalanobis distance, the augmented samples of each batch can vary within the distribution range of the original training data to maintain the semantics.

The overall loss function of the continuous prompt tuning stage is defined as:

$$\mathcal{L}_{PT} = \mathcal{L}_{cos} + \alpha \times \mathcal{L}_{mahal}, \quad (4)$$

where α is a hyperparameter that adjusts the weight of $\mathcal{L}_{mahal}$. In the gradient descent process of $\mathcal{L}_{PT}$, we only adjust the parameters of the generator $G(\cdot)$ and fix the PLMs' parameters to obtain the continuous prompt embeddings that further amplifies the bias between different sensitive attributes.

Algorithm 1: Proposed CCPA framework.

```
Input: Pre-processed training corpus $\mathcal{S}$, PLM
       encoder $E(\cdot)$, Initial prompt generator $G(\cdot)$,
       Prompt template $T$, Hyperparameter $\alpha, \beta, \tau$.
while stage 1 do
    Apply $T$ to $\forall(s_i, s'_i) \in \mathcal{S}$ to obtain $(\hat{s}_i, \hat{s}'_i)$;
    Obtain $(\mathbf{z}_i, \mathbf{z}'_i) = (E(\hat{s}_i), E(\hat{s}'_i))$;
    Replace $\{\mathbf{h}_1, \mathbf{h}_2, \cdots, \mathbf{h}_m\}$ encoded by $G(\cdot)$ in
      the corresponding position in $(\mathbf{z}_i, \mathbf{z}'_i)$;
    Calculate $\mathcal{L}_{cos}$ and $\mathcal{L}_{mahal}$ with $\{(\mathbf{z}_i, \mathbf{z}'_i)\}_{i=1}^{n}$;
    Update $G(\cdot)$'s parameters following Equation 4;
end
while stage 2 do
    Mask $\forall(s_i, s'_i)$ randomly with a 15% probability;
    Obtain $\{(\mathbf{z}_i, \mathbf{z}'_i)\}_{i=1}^{n}$ using $E(\cdot)$ and $G(\cdot)$;
    Calculate $\mathcal{L}_{nce}$ and $\mathcal{L}_{mlm}$ and update $E(\cdot)$'s
      parameters following Equation 6.
end
```

2.3 Fine-Tuning with Contrastive Learning

We then use contrastive learning to mitigate the social bias in PLMs' encoding for different demographic groups. Contrastive learning (Yang et al., 2019) is a task-agnostic self-supervision method that learns data features by minimizing contrastive loss to maximize the similarity of the representation vectors of positive sample pairs (Das et al., 2022). Specifically, we encourage as much consistency as possible among representations of different sensitive attributes by maximizing the similarity of the augmented counterfactual pairs. Noise Contrast Estimation (Gutmann and Hyvärinen, 2010) is usually used as a contrastive loss function, given an augmented sample pair of a batch $\{(\hat{s}_i, \hat{s}'_i)\}_{i=1}^{n}$, which is defined as follows:

$$\mathcal{L}_{nce} = \frac{1}{n} \sum_{i=1}^{n} \log \frac{e^{sim(\mathbf{z}_i, \mathbf{z}'_i)/\tau}}{\frac{1}{n}\sum_{j=1}^{n} e^{sim(\mathbf{z}_i, \mathbf{z}_j)/\tau}}, \quad (5)$$

where $(\mathbf{z}_i, \mathbf{z}'_i) = (E(\hat{s}_i), E(\hat{s}'_i))$, τ is a temperature hyperparameter and $sim(\cdot, \cdot)$ denotes the similarity function usually using cosine similarity. During training, we only fine-tune the PLMs' parameters and fix the embedding of continuous prompts. By maximizing $\mathcal{L}_{nce}$, differences in the encoding of PLM outputs specific to different demographic groups are eliminated, resulting in representations independent of sensitive attributes.

Considering that the attenuation of biases towards encoding may affect PLMs' language modeling capability, we add a Masking Language Modeling (MLM) loss during the fine-tuning stage to aid PLM training (He et al., 2022). Following previous work (Devlin et al., 2019), we randomly mask tokens in training texts with a 15% probability.[2] Our objective is to train the encoder to predict the masked tokens through contextual semantics, thereby preserving the language modeling capability of PLMs. The overall loss function in the fine-tuning stage is defined as follows:

$$\mathcal{L}_{FT} = \mathcal{L}_{nce} + \beta \times \mathcal{L}_{mlm}, \tag{6}$$

where β is a hyperparameter that controls the weight of $\mathcal{L}_{mlm}$. Our overall algorithm is given in Algorithm 1.

3 Experiments

In this section, we conduct experiments to evaluate the performance of CCPA, in order to answer the following three research questions.

Q1. How effective is CCPA in mitigating social biases in PLMs' encoding?

Q2. How does each component affect CCPA?

Q3. Will CCPA preserve the language modeling capability of PLMs?

3.1 Experimental Setup

3.1.1 Attribute Word List & Datasets

Following (Bolukbasi et al., 2016; Liang et al., 2020; Cheng et al., 2021; He et al., 2022), our gender attribute word list is set to:
{*MALE, FEMALE*}={*(man, woman), (boy, girl), (he, she), (father, mother), (son, daughter), (guy, gal), (male, female), (his, her), (himself, herself), (John, Mary)*}.

Following (Liang et al., 2020; Cheng et al., 2021), we select five real-world datasets as the initial training corpus, which are Stanford Sentiment Treebank (Socher et al., 2013), POM (Park et al., 2014), WikiText-2 (Merity et al., 2017), Reddit (Völske et al., 2017) and MELD (Poria et al., 2019) respectively. We set the maximum sentence length to 100, and the pre-processed training corpus contained 10,510 sentences.

3.1.2 Baselines & Implementation Details

We select seven recent task-agnostic debiasing models as baselines. **CDA** (Zmigrod et al., 2019), **Dropout** (Webster et al., 2020), **Sent-Debias** (Liang et al., 2020), **FairFil** (Cheng et al., 2021), **INLP** (Ravfogel et al., 2020) and **MABEL** (He et al., 2022) apply counterfactual data augmentation to sentence-level debiasing, where **FairFil** and **MABEL** adopt the contrastive learning framework training model. **Auto-Debias** (Guo et al., 2022) directly uses the attribute word list and the stereotype words list as the training corpus.

We perform the main experiments on BERT (Devlin et al., 2019) and compare CCPA to all baseline models. We also test debiasing performance on DistilBERT (Sanh et al., 2019) and ELEATRA (Clark et al., 2020). All checkpoints use *bert-base-uncased*, *distilbert-base-uncased*, and *google/electra-base-generator* implemented by Huggingface Transformers library (Wolf et al., 2020). In the continuous prompt tuning stage, the learning rate is set to $1e^{-5}$, the batch size is set to 64 and $\alpha = 0.005$. Following P-Tuning (Liu et al., 2021b), the virtual tokens template of continuous prompts is denoted as a triplet with the length of each element selected on $\{1, 2, 3\}$. In the fine-tuning stage, the learning rate is set to $1e^{-4}$. The batch size is set to 32, $\beta = 1$ and $\tau = 1$. We report the average of the results of three runs over 20 epochs.

To compare the baseline models more fairly, we apply the same attribute word lists and training datasets to CDA and Dropout as CCPA. The implementation codes for CDA, Dropout, Sent-Debias, and INLP are provided by (Meade et al., 2022), and the implementation codes for FairFil and Auto-Debias are provided by the authors. For MABEL, we report the results from its original paper.

3.2 Evaluation Metrics

We measure debiasing performance using the common three internal bias evaluation metrics and two external bias evaluation metrics.

3.2.1 Internal Bias Evaluation Metrics

Sentence Encoder Association Test (SEAT) (May et al., 2019) uses sentence templates to evaluate the association between different sensitive attribute demographic and target concepts. Given the attribute word lists $\mathcal{A}$ and $\mathcal{B}$, the target words lists $\mathcal{X}, \mathcal{Y}$. The results are presented by effect size, defined as:

$$d = \frac{\mu(\{s(x, \mathcal{A}, \mathcal{B})\}) - \mu(\{s(y, \mathcal{A}, \mathcal{B})\})}{\sigma(\{s(t, \mathcal{X}, \mathcal{Y})\}_{t \in \mathcal{A} \cup \mathcal{B}})}, \tag{7}$$

where $x \in \mathcal{X}$ and $y \in \mathcal{Y}$, $\mu(\cdot)$ is the mean function and $\sigma(\cdot)$ is the standard deviation. And $s(w, \mathcal{A}, \mathcal{B})$

[2] In practice, the chosen masked token has an 80% chance of being masked, a 10% chance of being replaced with another word, and a 10% chance of remaining unchanged.

Model \ Metric	SEAT-6	SEAT-6b	SEAT-7	SEAT-7b	SEAT-8	SEAT-8b	Avg.	LM	SS	ICAT	CrowS
BERT	0.932	0.090	-0.124	0.937	0.783	0.858	0.621	84.17	60.28	66.86	57.86
+CDA	0.596	-0.103	-0.236	0.800	0.394	0.734	0.477	85.47	58.69	70.63	55.35
+Dropout	0.912	0.121	0.321	0.857	0.777	0.867	0.642	85.42	60.11	68.16	55.35
+Sent-Debias	0.336	-0.314	-0.624	**0.514**	0.391	0.436	0.436	**85.60**	59.05	70.11	42.14
+FairFil	0.683	-0.140	-0.616	0.839	**0.049**	-0.501	0.471	48.78	**46.44**	45.31	62.89
+Auto-Debias	0.373	**-0.056**	0.745	1.175	0.856	0.823	0.671	81.76	57.33	69.78	52.83
+INLP	0.619	-0.226	0.326	0.591	0.430	0.549	0.457	82.69	58.09	69.31	50.94
+MABEL	0.664	0.167	0.479	0.647	0.465	0.570	0.499	84.80	56.92	73.07	**50.76**
+CCPA (Ours)	**0.181**	-0.317	**0.104**	0.633	0.142	**0.115**	**0.249**	84.44	56.61	**73.28**	51.57
DistilBERT	1.380	0.446	-0.179	1.242	0.837	1.217	0.883	**84.75**	60.52	66.93	59.75
+CCPA (Ours)	**0.409**	**-0.024**	**0.138**	**-0.029**	**-0.029**	**0.283**	**0.152**	81.91	**56.47**	**71.30**	**50.31**
ELEATRA	0.820	0.036	1.180	1.007	0.782	0.958	0.797	**85.12**	58.15	71.24	52.83
+CCPA (Ours)	**0.251**	**0.012**	**0.647**	**0.437**	**0.720**	**0.460**	**0.421**	84.63	**52.97**	**79.61**	**49.06**

Table 1: Gender debiasing results of SEAT, StereoSet, and CrowSPairs on BERT, DistilBERT, and ELEATRA. The best result is indicated in **bold**. We report CCPA results with a continuous prompt template $(1, 1, 1)$. The closer the effect size is to 0 and the closer SS is to 50%, the higher the fairness; the higher the LM and ICAT, the better.

is the bias degree defined as: $s(w, \mathcal{A}, \mathcal{B}) = \mu(cos(w, a)) - \mu(cos(w, b))$.

The gender-specific subsets of SEAT are 6, 6b, 7, 7b, 8, and 8b. We report the effect size of debiasing models on each subset and the average value of the absolute value of the six subsets, respectively.

StereoSet (Nadeem et al., 2021) uses the fill-in-the-blank template to investigate the stereotype association of PLM. The Language Modeling Score (LM) is the percentage of stereotype or anti-stereotype words selected by the model based on incomplete contextual sentences. The Stereotype Score (SS) is the percentage of models that choose stereotypes over anti-stereotypes. The Idealized Context Association Test (ICAT) is a comprehensive evaluation index of LM and SS.

Crowdsourced Stereotype Pairs (CrowS-Pairs) (Nangia et al., 2020) is a dataset containing pairs of stereotype sentences and anti-stereotype sentences. We report the ratio of mask token probabilities assigned to stereotype sentences rather than anti-stereotype sentences, denoted using CrowS.

3.2.2 External Bias Evaluation Metrics

Bias-in-Bios (De-Arteaga et al., 2019) is a biography dataset in which each sample is labeled with gender (male or female) and occupation (28 categories). We fine-tune the debiased model on the training set with the goal of predicting occupations. Overall Accuracy result is used to measure task precision, and individual Accuracy results for male and female are used to measure gender fairness. Furthermore, we report the gap between the true positive rates of the male prediction results and the female prediction results denotes as GAP_{TPR}, as well as the root mean square of the true positive rates difference for each category denotes as GAP_{RMS}. The closer their score is to 0, the better. They are defined as follows:

$$GAP_{TPR} = |TPR_M - TPR_F|, \tag{8}$$

$$GAP_{RMS} = \sqrt{\frac{1}{|C|}\sum_{y\in C}(GAP_{TPR,y})^2}. \tag{9}$$

Bias-NLI (Dev et al., 2020) fills gender words and occupation words with stereotypes into sentence templates to form sentence pairs, and the training goal is to inference whether the sentence pair is neutral or not. It defines three metrics to reflect the fairness of the model: 1) Net Neutral (NN), the average probability of neutral labels across all sentence pairs; 2) Fraction Neutral (FN), the proportion of sentence pairs marked as neutral; 3) Threshold:τ (T:τ), The fraction of samples with neutral probability above τ is reported.

3.3 Debiasing Performance Analysis

3.3.1 Internal Debiasing Results

Table 1 shows the experimental results of three bias evaluation metrics for CCPA and baseline models on BERT, DistilBERT, and ELEATRA. We also report results for biased BERT, DistilBERT, and ELEATRA as references. The results show that CCPA achieves a better balance between PLMs' fairness and language modeling capability than the baseline models.

For BERT, CCPA reduces the average effect size from 0.621 to 0.249, increases ICAT from 66.86 to

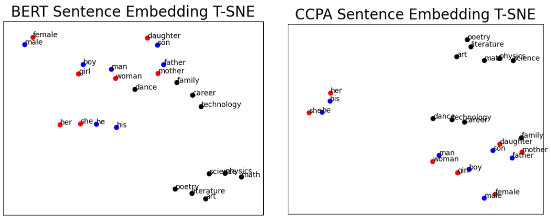

Figure 3: T-SNE plots of sentence level representations encoded by BERT and CCPA. We use target words and their sentence templates in SEAT.

73.28, and reduces CrowS from 57.86 to 51.57. Our method has achieved optimal results in the three test subsets of SEAT 6, 7, 8b and the average effect size, and has also been greatly improved in the other test subsets. The results on StereoSet show that CCPA does not weaken BERT's language modeling ability but slightly improves it. Although LM and SS do not achieve optimal results, our comprehensive index ICAT is better than other models. Both FairFil and MABEL are biased by contrastive learning, but their overall performance is not ideal. Although FairFil is outstanding in terms of SS performance, it seriously damages BERT's language modeling ability, possibly because it only considers sentence-level representation and does not retain token-level encoding ability. MABEL achieves promising results on StereoSet and CrowS-Pairs, but its SEAT results must be improved. Regarding overall performance, CCPA outperforms other contrastive learning frameworks, demonstrating that our adversarial training inspired approach can improve the model's learning ability by increasing the complex information in the model.

For DistilBERT, CCPA decreases the average effect size from 0.883 to 0.152 and improves ICAT from 66.93 to 71.30. Our model gets excellent experimental results on most test subsets of SEAT and reaches an almost ideal 50.31% result on CrowS-Pairs. LM score decreases, and we analyze that the semantic information of the original representation is affected by too much debiasing.

For ELEATRA, which does not belong to the bert-series PLM, the debiasing effect of CCPA is equally significant, and the experimental results are fairer than the original ELEATRA on all three intrinsic metrics. In detail, CCPA reduced the average effect size from 0.797 to 0.421, increases ICAT by 8.37% without significantly decreasing LM score, and reduces CrowS score by 1.89%.

Model	Acc.	Acc.	Acc.	GAP	GAP
	(All)	(M)	(F)	TPR	RMS
BERT	84.14	84.69	83.50	1.189	0.144
+INLP	70.50	-	-	-	**0.067**
+Sent-Debias	83.56	84.10	82.92	1.180	0.144
+FairFil	83.18	83.52	82.78	0.746	0.142
+MABEL	84.85	84.92	84.34	0.599	0.132
+CCPA (Ours)	**85.65**	**85.41**	**85.95**	**0.544**	0.121

Table 2: Results on Bias-in-Bios. The best result is indicated in **bold**. The sub-optimal result is indicated in underline. '-' means not reported.

Model	NN	FN	T:0.5	T:0.7
BERT	0.799	0.879	0.874	0.798
+Sent-Debias	0.793	0.911	0.897	0.788
+FairFil	0.829	0.883	0.846	0.845
+MABEL	**0.900**	**0.977**	**0.974**	**0.935**
+CCPA (Ours)	0.883	0.932	0.929	0.878

Table 3: Results on Bias-NLI. The best result is indicated in **bold**. The sub-optimal result is indicated in underline.

We also perform a small qualitative study by visualizing t-SNE plots of sentence embedding. As can be seen from Figure 3, in BERT, male attribute words are more inclined to target words in the technical field (such as *career* or *science*) in the embedded space, while female attribute words are more inclined to target words in the humanities (such as *family* or *poetry*). After using CCPA to debias, it is observed that gender-attribute words are pulled closer together and away from neutral words in the representational space.

3.3.2 External Debiasing Results

We fine-tune the debiased BERT on two downstream tasks Bias-in-Bios and Bias-NLI to verify the effect of CCPA on external debiasing, and the results are shown in Tables 2 and 3. All our experimental setups are consistent with MABEL, and all the results reported in the table for the baseline models are from MABEL.

On the Bias-in-Bios task as shown in Table 2, CCPA not only achieves the optimal results on task accuracy, but also performs the best on all gender fairness metrics except GAP_{RMS}. Although INLP obtains the best score on the GAP_{RMS} metric, its task accuracy is clearly impaired from the reported results. Compared to all baselines, CCPA achieves the best overall debiasing performance while preserving the model's prediction performance on downstream tasks.

On the Bias-NLI task as shown in Table 3, CCPA

BERT+		SEAT-6	SEAT-6b	SEAT-7	SEAT-7b	SEAT-8	SEAT-8b	Avg.	LM	SS	ICAT	CrowS
$T_{(1,1,1)}$	CCPA	0.181	-0.317	**0.104**	0.633	**0.142**	0.115	**0.249**	**84.44**	56.61	**73.28**	51.57
	CCPA$^-$	-0.198	**-0.100**	-0.162	**-0.309**	0.535	0.243	0.258	79.34	57.31	67.75	47.17
	CCPA*	**0.044**	-0.295	-0.340	0.425	-0.400	**0.091**	0.266	78.94	**56.13**	69.26	**50.31**
$T_{(2,2,2)}$	CCPA	0.126	-0.135	-0.379	0.144	0.416	**0.056**	**0.209**	**82.40**	**55.36**	**73.56**	51.57
	CCPA$^-$	0.264	**0.065**	**-0.372**	**0.127**	0.150	0.481	0.243	79.23	57.44	67.44	48.43
	CCPA*	**0.0918**	-0.178	0.509	0.311	**0.144**	0.271	0.251	79.84	54.94	71.95	**50.94**
$T_{(3,3,3)}$	CCPA	**0.034**	**-0.006**	**0.193**	0.278	-0.189	0.346	**0.174**	**82.62**	**54.80**	**74.68**	**49.06**
	CCPA$^-$	0.149	0.037	-0.826	**-0.106**	-0.124	**-0.303**	0.258	79.18	59.63	63.92	43.40
	CCPA*	0.119	-0.131	-0.334	0.225	**-0.098**	-0.365	0.212	79.95	56.93	68.87	**50.94**
NO$_{prompt}$		0.325	0.186	0.342	0.535	0.144	0.553	0.347	81.32	54.82	73.48	60.38
NO$_{prompt+mask}$		0.730	-0.012	-0.185	-0.530	0.927	-0.158	0.424	61.85	53.98	56.92	34.59

Table 4: Gender debiasing results of SEAT, StereoSet and CrowSPairs on BERT. The best result is indicated in **bold**. The closer the effect size is to 0 and the closer SS is to 50%, the better; the higher the LM and ICAT, the better.

achieves sub-optimal results on all the metrics. It is worth stating that MABEL is a debiasing method trained on the NLI task, which we analyze as the main reason for its most outstanding performance. Even so, the strong debiasing effect shown by CCPA on task Bias-NLI is heartening.

The results of the internal debiasing experiment and the external debiasing experiment show that our proposed CCPA has outstanding performance in mitigating gender bias in PLMs' encoding. CCPA has an efficient debiasing performance, which answers the first question (**Q1**) proposed at the beginning of this section.

3.4 Ablation Analysis

We conduct ablation experiments on BERT to investigate how each component affects CCPA performance. The results are shown in Table 4.

$T_{(1,1,1)}$ indicates that the continuous prompt template is a triplet with one virtual token for each element, i.e., the length of prompts is 3. By analogy, $T_{(2,2,2)}$ and $T_{(3,3,3)}$ represent prompt templates of lengths 6 and 9. The purpose of this setting is to make it easier to observe the effect of the prompts' length on the model. In the experimental group of each template, we compare three versions of CCPA: the original CCPA, the version without $\mathcal{L}_{mlm}$ represented as CCPA$^-$ and the version without $\mathcal{L}_{mahal}$ represented as CCPA*. In addition, we have experimented with both CCPA without prompts and CCPA without prompts and $\mathcal{L}_{mlm}$.

It is observed from the experimental results that the debiasing ability of CCPA increases with the rise of the template's length. This indicates that longer continuous prompt embeddings bring more difficult information to the model, thus increasing the debiasing effort. However, more extended templates can cause the original sentence semantics to be broken and thus weaken PLM's language modeling capability. In each experimental group, both CCPA$^-$ and CCPA* show a decrease in the results of the three evaluation metrics compared to CCPA. This phenomenon verifies that both MLM-assisted loss and Mahalanobis distance constraint benefit CCPA. Overall, MLM has a greater influence, especially on SS and CrowS, which may be because random mask tokens train encoders to retain token-level semantic information.

In addition, the results of NO$_{prompt}$ verify that continuous prompts play an essential role in CCPA. NO$_{prompt+mask}$ tests the effect of fine-tuning PLMs based solely on contrastive learning. Unsurprisingly, the performance on all indexes could be better. The results of NO$_{prompt}$ and NO$_{prompt+mask}$ again reflect our method's effectiveness. The ablation studies answer our second question (**Q2**) by exploring the role played by each component of the CCPA.

3.5 Language Modeling Capability Analysis

We perform experiments on nine natural language understanding tasks of the GLUE benchmark to verify the language modeling capability of CCPA on downstream tasks. In task-specific fine-tuning, we set the learning rate to $2e-5$ and the batch size to 32 for all models.

As in Table 5, CCPA's performance in 9 tasks is comparable to that of the original BERT, and the average results are almost equivalent to BERT's. CCPA also shows similar performance on DistilBERT, indicating that our model is effective on other models besides BERT. Combined with the LM score in Table 1, the experiment shows that CCPA can debias without damaging the language modeling capability of PLMs, thus answering the third research question (**Q3**).

Model	CoLA	MNLI	MRPC	QNLI	QQP	RTE	SST	STS-B	WNLI	Average
BERT	56.78	84.76	**89.54**	91.51	88.06	64.62	**93.35**	88.24	**56.34**	79.24
+CDA	2.07	84.84	81.22	84.84	87.85	47.29	92.32	40.83	43.66	62.77
+Dropout	2.07	84.78	81.22	91.49	88.02	47.29	92.09	40.87	43.66	63.50
+Sent-Debias	55.72	**84.94**	88.81	91.54	87.88	63.90	93.12	88.23	**56.34**	78.94
+FairFil	55.72	84.85	88.33	**91.84**	87.43	64.98	93.12	88.55	50.7	78.39
+Auto-Debias	57.01	84.91	88.54	91.65	87.92	64.62	92.89	88.43	40.85	77.42
+INLP	56.50	84.78	89.23	91.38	87.94	**65.34**	92.66	88.73	54.93	79.05
+MABEL	**57.80**	84.50	85.00	91.60	**88.10**	64.30	92.20	**89.20**	-	**81.59**
+CCPA	55.91	84.73	88.65	91.42	87.98	64.93	93.09	88.44	55.66	78.98
DistilBERT	**47.93**	82.01	**88.47**	**88.61**	86 68	**58.84**	90.71	**86.26**	**56.34**	**76.21**
+CCPA	46.73	**82.53**	86.99	87.76	**86.85**	56.26	**90.83**	85.89	55.93	75.53

Table 5: Experimental results of GLUE tasks on BERT and DistilBERT. We report Matthew's correlation for CoLA, the Spearman correlation for STS-B, and the F1 score for MRPC and QQP. Other tasks are reported for the accuracy. The best result is indicated in **bold**. '-' means not reported in MABEL.

4 Related Work

We divide debiasing methods into two categories based on the debiasing strategy: task-specific methods and task-agnostic methods.

4.1 Task-Specific Methods

Task-specific methods adopt the strategy of debiasing in the fine-tuning stage of the downstream task, of which the downstream task is known (Han et al., 2021; Chi et al., 2022). One representative work is INLP (Ravfogel et al., 2020, 2022), which repeatedly trains a linear classifier that predicts the target concept, and then projects the representation into the null space of the classifier's weight matrix to remove the representation bias. Contrastive learning is proposed to mitigate bias in classifier training (Shen et al., 2021). It encourages instances sharing the same class labels to have similar representations while ensuring that protected attributes have different distributions. These methods use attribute words to label training data without CDA. However, they are biased towards specific downstream tasks and cannot be applied to other tasks in general. When training data change, task-specific methods are difficult to transfer to new tasks.

4.2 Task-Agnostic Methods

Task-agnostic methods adopt the strategy of debiasing representation or processing unbalanced data before the downstream task, and they can be applied to any downstream task (Dev et al., 2020, 2021). Most of these methods apply counterfactual data augmentation to augment the unbalanced corpus and then debias the augmented text information. Counterfactual data augmentation (Lu et al., 2020) is a general approach to augment corpora through causal intervention and has since been widely used to mitigate social biases. Different variants of counterfactual data augmentation have been proposed, such as Sent-Debias (Liang et al., 2020), FairFil (Cheng et al., 2021), MABEL (He et al., 2022), to name a few examples.

Task-agnostic methods primarily use the CDA to balance the training corpus by constructing counterfactual pairs specific to different demographic groups. However, simply applying CDA to the original corpus makes minor changes, constraining the representation space to a narrow range. This makes the model easily fit the differences between counterfactual pairs, weakening the debiasing ability. Unlike existing CDA methods, we train a generator that encodes continuous prompts before fine-tuning PLM. The goal is to widen the representation distance between different groups to increase the difficulty of the model-learning process.

5 Conclusions

Inspired by adversarial training, we propose CCPA, a two-stage debiasing model that combines contrastive learning with continuous prompts. In the continuous prompt tuning stage, we train a generator encoding continuous prompt embeddings to increase the representative distance between counterfactual pairs. In the fine-tuning stage, we use contrastive learning to reduce the representation distance between the augmented sample pairs. By increasing the difficulty of the training process, CCPA enables PLMs to learn a stronger debiasing ability. Extensive experiments on BERT and DistilBERT show that CCPA effectively reduces social bias in PLM representation while retaining language modeling capability.

Limitations

In this work, we focus on debiasing the gender bias for PLMs. In the future, we will try to mitigate social biases other than gender, such as race and religion. In addition, we also plan to extend our debiasing method to more language models, such as Natural Language Generation (NLG) models.

Ethics Statement

This paper has been thoroughly reviewed for ethical considerations and has been found to be in compliance with all relevant ethical guidelines. The paper does not raise any ethical concerns and is a valuable contribution to the field.

Acknowledgments

We express gratitude to the anonymous reviewers for their hard work and kind comments. The work was supported in part by the National Natural Science Foundation of China (No.62272191, No.61976102), the Science and Technology Development Program of Jilin Province (No.20220201153GX), the Interdisciplinary and Integrated Innovation of JLU (No.JLUXKJC2020207), and the Graduate Innovation Fund of Jilin University (No.2022214).

References

Emily M. Bender and Batya Friedman. 2018. Data statements for natural language processing: Toward mitigating system bias and enabling better science. *Trans. Assoc. Comput. Linguistics, TACL*, 6:587–604.

Tolga Bolukbasi, Kai-Wei Chang, James Y. Zou, Venkatesh Saligrama, and Adam Tauman Kalai. 2016. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. In *Proceedings of the Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems,NeurIPS*, pages 4349–4357.

Hao Chen, Rui Xia, and Jianfei Yu. 2021. Reinforced counterfactual data augmentation for dual sentiment classification. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP*, pages 269–278. Association for Computational Linguistics.

Pengyu Cheng, Weituo Hao, Siyang Yuan, Shijing Si, and Lawrence Carin. 2021. Fairfil: Contrastive neural debiasing method for pretrained text encoders. In *Proceedings of the 9th International Conference on Learning Representations, ICLR*. OpenReview.net.

Jianfeng Chi, William Shand, Yaodong Yu, Kai-Wei Chang, Han Zhao, and Yuan Tian. 2022. Conditional supervised contrastive learning for fair text classification. *CoRR*, abs/2205.11485.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: pre-training text encoders as discriminators rather than generators. In *Proceedings of the 8th International Conference on Learning Representations, ICLR*. OpenReview.net.

Sarkar Snigdha Sarathi Das, Arzoo Katiyar, Rebecca J. Passonneau, and Rui Zhang. 2022. Container: Few-shot named entity recognition via contrastive learning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics, ACL*, pages 6338–6353. Association for Computational Linguistics.

Maria De-Arteaga, Alexey Romanov, Hanna M. Wallach, Jennifer T. Chayes, Christian Borgs, Alexandra Chouldechova, Sahin Cem Geyik, Krishnaram Kenthapadi, and Adam Tauman Kalai. 2019. Bias in bios: A case study of semantic representation bias in a high-stakes setting. In *Proceedings of the Conference on Fairness, Accountability, and Transparency, FAT*, pages 120–128. ACM.

Pieter Delobelle, Ewoenam Kwaku Tokpo, Toon Calders, and Bettina Berendt. 2022. Measuring fairness with biased rulers: A comparative study on bias metrics for pre-trained language models. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL*, pages 1693–1706. Association for Computational Linguistics.

Sunipa Dev, Tao Li, Jeff M. Phillips, and Vivek Srikumar. 2020. On measuring and mitigating biased inferences of word embeddings. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence*, pages 7659–7666. AAAI Press.

Sunipa Dev, Tao Li, Jeff M. Phillips, and Vivek Srikumar. 2021. Oscar: Orthogonal subspace correction and rectification of biases in word embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP*, pages 5034–5050. Association for Computational Linguistics.

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, NAACL*, pages 4171–4186. Association for Computational Linguistics.

Yue Guo, Yi Yang, and Ahmed Abbasi. 2022. Autodebias: Debiasing masked language models with automated biased prompts. In *Proceedings of the 60th*

Annual Meeting of the Association for Computational Linguistics, ACL, pages 1012–1023. Association for Computational Linguistics.

Michael Gutmann and Aapo Hyvärinen. 2010. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the 13th International Conference on Artificial Intelligence and Statistics, AISTATS*, volume 9 of *JMLR Proceedings*, pages 297–304. JMLR.org.

Xudong Han, Timothy Baldwin, and Trevor Cohn. 2021. Diverse adversaries for mitigating bias in training. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, EACL*, pages 2760–2765. Association for Computational Linguistics.

Jacqueline He, Mengzhou Xia, Christiane Fellbaum, and Danqi Chen. 2022. MABEL: Attenuating gender bias using textual entailment data. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP*.

Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP*, pages 4582–4597. Association for Computational Linguistics.

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

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2021a. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *CoRR*, abs/2107.13586.

Xiao Liu, Yanan Zheng, Zhengxiao Du, Ming Ding, Yujie Qian, Zhilin Yang, and Jie Tang. 2021b. GPT understands, too. *CoRR*, abs/2103.10385.

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

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, NAACL-HLT*, pages 622–628. 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, ACL*, pages 1878–1898. Association for Computational Linguistics.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. 2017. Pointer sentinel mixture models. In *Proceedings of the 5th International Conference on Learning Representations, ICLR*. OpenReview.net.

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

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

Sunghyun Park, Han Suk Shim, Moitreya Chatterjee, Kenji Sagae, and Louis-Philippe Morency. 2014. Computational analysis of persuasiveness in social multimedia: A novel dataset and multimodal prediction approach. In *Proceedings of the 16th International Conference on Multimodal Interaction, ICMI*, pages 50–57. ACM.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL*, pages 2227–2237. Association for Computational Linguistics.

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

Soujanya Poria, Devamanyu Hazarika, Navonil Majumder, Gautam Naik, Erik Cambria, and Rada Mihalcea. 2019. MELD: A multimodal multi-party dataset for emotion recognition in conversations. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL*, pages 527–536. Association for Computational Linguistics.

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*, pages 7237–7256. Association for Computational Linguistics.

Shauli Ravfogel, Michael Twiton, Yoav Goldberg, and Ryan Cotterell. 2022. Linear adversarial concept erasure. In *Proceedings of the International Conference on Machine Learning, ICML*, volume 162 of *Proceedings of Machine Learning Research*, pages 18400–18421. PMLR.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of BERT: smaller, faster, cheaper and lighter. *CoRR*, abs/1910.01108.

Timo Schick and Hinrich Schütze. 2020. Few-shot text generation with pattern-exploiting training. *CoRR*, abs/2012.11926.

Timo Schick and Hinrich Schütze. 2021. Exploiting cloze-questions for few-shot text classification and natural language inference. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, EACL 2021, Online, April 19 - 23, 2021*, pages 255–269. Association for Computational Linguistics.

Aili Shen, Xudong Han, Trevor Cohn, Timothy Baldwin, and Lea Frermann. 2021. Contrastive learning for fair representations. *CoRR*, abs/2109.10645.

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

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Y. Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP*, pages 1631–1642. ACL.

Michael Völske, Martin Potthast, Shahbaz Syed, and Benno Stein. 2017. Tl;dr: Mining reddit to learn automatic summarization. In *Proceedings of the Workshop on New Frontiers in Summarization, NFiS@EMNLP*, pages 59–63. Association for Computational Linguistics.

Yufei Wang, Can Xu, Qingfeng Sun, Huang Hu, Chongyang Tao, Xiubo Geng, and Daxin Jiang. 2022. Promda: Prompt-based data augmentation for low-resource NLU tasks. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics, ACL*, pages 4242–4255. Association for Computational Linguistics.

Kellie Webster, Xuezhi Wang, Ian Tenney, Alex Beutel, Emily Pitler, Ellie Pavlick, Jilin Chen, and Slav Petrov. 2020. Measuring and reducing gendered correlations in pre-trained models. *CoRR*, abs/2010.06032.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi 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 M. 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, EMNLP*, pages 38–45. Association for Computational Linguistics.

Zonghan Yang, Yong Cheng, Yang Liu, and Maosong Sun. 2019. Reducing word omission errors in neural machine translation: A contrastive learning approach. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL*, pages 6191–6196. Association for Computational Linguistics.

Jieyu Zhao, Tianlu Wang, Mark Yatskar, Ryan Cotterell, Vicente Ordonez, and Kai-Wei Chang. 2019. Gender bias in contextualized word embeddings. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL*, pages 629–634. Association for Computational Linguistics.

Zexuan Zhong, Dan Friedman, and Danqi Chen. 2021. Factual probing is [MASK]: learning vs. learning to recall. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT*, pages 5017–5033. Association for Computational Linguistics.

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

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
Limitations

□ A2. Did you discuss any potential risks of your work?
Not applicable. Left blank.

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Abstract; 5 Conclusions

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☒ Did you use or create scientific artifacts?

Left blank.

□ B1. Did you cite the creators of artifacts you used?
No response.

□ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
No response.

□ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
No response.

□ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
No response.

□ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
No response.

□ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
No response.

C ☑ Did you run computational experiments?

3 Experiments

☒ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Our model uses the standard pre-trained language model as the backbone network, and the number of parameters and consumption are equivalent to that of the pre-trained language model. This part is not the focus of our discussion and is not specifically mentioned.

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
3.1.2 Baselines & Implementation Detai

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
3.1.2 Baselines & Implementation Detai

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
3.1.2 Baselines & Implementation Detai

D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

□ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
No response.

□ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
No response.

□ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
No response.

□ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
No response.

□ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
No response.