Artificial Intelligence 332 (2024) 104143

Contents lists available at ScienceDirect

Artificial Intelligence

journal homepage: www.elsevier.com/locate/artint

Mitigating social biases of pre-trained language models via contrastive self-debiasing with double data augmentation

Yingji Li [a], Mengnan Du [b], Rui Song [c], Xin Wang [c], Mingchen Sun [a], Ying Wang [a,d,*]

[a] College of Computer Science and Technology, Jilin University, Changchun, 130012, China
[b] Department of Data Science, New Jersey Institute of Technology, Newark, USA
[c] School of Artificial Intelligence, Jilin University, Changchun, 130012, China
[d] Key Laboratory of Symbol Computation and Knowledge Engineering (Jilin University), Ministry of Education, Changchun, 130012, China

ARTICLE INFO

Keywords:
Pre-trained language models
Social bias
Contrastive learning
Prompt learning
Data augmentation

ABSTRACT

Pre-trained Language Models (PLMs) have been shown to inherit and even amplify the social biases contained in the training corpus, leading to undesired stereotype in real-world applications. Existing techniques for mitigating the social biases of PLMs mainly rely on data augmentation with manually designed prior knowledge or fine-tuning with abundant external corpora to debias. However, these methods are not only limited by artificial experience, but also consume a lot of resources to access all the parameters of the PLMs and are prone to introduce new external biases when fine-tuning with external corpora. In this paper, we propose a Contrastive Self-Debiasing Model with Double Data Augmentation (named CD^3) for mitigating social biases of PLMs. Specifically, CD^3 consists of two stages: double data augmentation and contrastive self-debiasing. First, we build on counterfactual data augmentation to perform a secondary augmentation using biased prompts that are automatically searched by maximizing the differences in PLMs' encoding across demographic groups. Double data augmentation further amplifies the biases between sample pairs to break the limitations of previous debiasing models that heavily rely on prior knowledge in data augmentation. We then leverage the augmented data for contrastive learning to train a plug-and-play adapter to mitigate the social biases in PLMs' encoding without tuning the PLMs. Extensive experimental results on BERT, ALBERT, and RoBERTa on several real-world datasets and fairness metrics show that CD^3 outperforms baseline models on gender debiasing and race debiasing while retaining the language modeling capabilities of PLMs.

1. Introduction

In recent years, large-scale Pre-trained Language Models (PLMs) have been widely studied in Natural Language Processing (NLP) field [1]. They have also been applied to real-world scenarios, such as the automatic resume filtering system [2], the US healthcare system [3], etc. These NLP systems require PLMs not only to have good language modeling capabilities, but also to pay attention to the fairness of the results. Recent researches have shown that PLMs such as BERT, can inherit and even amplify implicit social biases in demographic groups contained in training corpora [4–6]. Unfair PLMs applied to real-world applications can lead to unpredictable

[*] Corresponding author.
E-mail addresses: yingji21@mails.jlu.edu.cn (Y. Li), mengnan.du@njit.edu (M. Du), songrui20@mails.jlu.edu.cn (R. Song), xinwang@jlu.edu.cn (X. Wang), mcsun20@mails.jlu.edu.cn (M. Sun), wangying2010@jlu.edu.cn (Y. Wang).

https://doi.org/10.1016/j.artint.2024.104143
Received 29 April 2023; Received in revised form 4 November 2023; Accepted 23 April 2024

Available online 26 April 2024

risks. For example, an automated resume filtering system could create human-like gender stereotypes, and it would tend to assign to men jobs that require logical abilities such as doctors or programmers, and to women jobs that require care such as nurses or babysitter [7]. Additionally, the US healthcare system may be racially biased, which calculates that black patients with the same risk level are sicker than white patients [3]. Therefore, mitigating social biases in PLMs' encoding has become a meaningful and challenging area of research.

Social biases in PLM-encoded representations are mainly due to the imbalanced number of training samples for different demographic groups [8,9]. An effective debiasing approach is to balance the training corpora using data augmentation [10–12] to increase the number of samples for different demographic groups. Most existing debiasing models augment samples with counterfactual data augmentation [13,14], which leverages a given set of sensitive attribute words (e.g., men/women, he/she) to generate texts targeted at different demographic groups. However, hand-designed sensitive attribute words heavily rely on the experience of researchers. As a result, the effect of data augmentation is limited by the quality of prior knowledge, which affects the debiasing performance.

In addition, there are other strategies to debias by changing the internal structure of the PLMs and then retraining the model or fine-tuning the parameters of the PLMs in downstream tasks. For example, Dropout [10] adds a dropout regularization during pre-training to intercept association between words in the context of attention mechanism within PLMs, and DebiasBERT [15] fine-tunes the parameters of the language model through a proposed debiasing loss function. However, these debiasing methods have drawbacks that cannot be ignored. Retraining-based methods are difficult to implement due to extremely high resource requirements. Fine-tuning based methods not only consume a lot of resources due to accessing the parameters of the entire model, but also are prone to introduce new biases when fine-tuning with abundant external corpora that affect the language modeling capabilities of PLMs on other tasks.

To address the above challenges, we propose a Contrastive Self-Debiasing Model with Double Data Augmentation (named CDDD, i.e., CD^3) to mitigate social biases in PLMs' encoding. Specifically, CD^3 consists of two stages: double data augmentations and contrastive self-debiasing. In the double data augmentation stage, we first augment the original training data with sensitive attribute words specific to different demographic groups and then perform a secondary data augmentation with the biased prompts for the positive sample pairs. We automatically search for biased prompts by maximizing the difference in PLMs' encoding across demographic groups. The obtained biased prompts are concatenated into the sentence with sensitive attribute words to amplify the bias between positive sample pairs. In the contrastive self-debiasing stage, the augmented samples are used for contrastive learning to mitigate the social biases implicit in PLMs' encoding. We train a plug-and-play adapter by minimizing the contrastive loss between different demographic groups. Our contributions are summarized as follows:

- We propose a double data augmentation strategy to further probe the biases between sample pairs using the automatically searched biased prompts, breaking the limitations of debiasing methods that heavily rely on prior knowledge in data augmentation.
- Our contrastive self-debiasing adapter can be widely applied to any PLM without changing the internal structure or parameters of the language model, which saves a lot of resources while preserving the language modeling capabilities of PLMs.
- Extensive experimental results on BERT [16], ALBERT [17], and RoBERTa [18] on several real-world datasets and fairness metrics show that CD^3 outperforms baseline models on gender debiasing and race debiasing while retaining the language modeling capabilities of PLMs.

2. Related work

2.1. Social bias of PLMs

In the research of NLP language models, it is often assumed that data are independent identically distributed. However, the data collected in the real world are essentially heterogeneous data with the behavior and characteristics of social subgroups [19,20]. Language models inherit human-like stereotypes from large unprocessed real corpora [21], and their encoding can even amplify biases in the data [22]. Social bias of PLMs usually refers to the model's differential treatment of a social subgroup, which can be roughly understood as two types [23]: intrinsic bias and extrinsic bias. Intrinsic bias is reflected in the model's dissimilarity in encoding different subgroups. Extrinsic bias is reflected in the model's different decisions for different subgroups in downstream tasks. According to the type of bias, the evaluation bias can be divided into two aspects [24]: intrinsic metrics and extrinsic metrics. Intrinsic metrics typical quantify bias by calculating the representational distance between subgroups and stereotypes. Extrinsic metrics are specific to downstream tasks such as sentiment classification and toxic text classification, and they measure the difference in the model's output predictions for different subgroups.

In this paper, we use two intrinsic metrics [25,26] and one extrinsic metric [27] in the experiments to comprehensively evaluate our proposed method. In order to compare with the general debiasing models, we choose two widely studied social biases, gender bias [28] and race bias [29] as experimental analysis cases.

2.2. Debiasing methods of PLMs

Researchers have made several attempts to mitigate the social biases in NLP systems. Existing debiasing techniques can be divided into three main categories based on the stage at which they are applied to language models [30,31]: pre-processing, in-processing, and post-processing.

2.2.1. Pre-processing

Pre-processing methods manipulate the training corpus before model training. One strategy is to handle the biases in the training data while keeping the model's subsequent training process unchanged. Amrhein et al. [32] use a machine translation model to round-trip translate biased text from truly fair text, creating a biased dataset to deal with languages that are morphologically more complex than English.

Since training samples collected from the real world for different demographic groups (e.g., male and female, black and white) are often imbalanced, another pre-processing strategy is to balance the number of training samples using data augmentation. Counterfactual data augmentation (CDA) [13,33] typically uses a given set of sensitive attribute words to generate samples for other demographic groups. Many debiasing studies have implemented CDA, such as the study conducted by Zhao et al. [34] which focuses on addressing gender-biased coreference resolution. This approach involves augmenting the original dataset by using a rule-based method to generate an auxiliary dataset in which male words are substituted with their female counterparts. MEBEL [35] applies CDA to samples from natural language inference (NLI) datasets, utilizing entailment labels to mitigate gender bias in context representations. TCWR [36] focuses on data augmentation of source and target phrases in machine translation. The specific procedure is to obtain the correspondence between the source and target phrase and then re-sample the source phrase from the masked language model and CDA with the causal model to obtain the aligned target phrase. Data augmentation techniques that solely rely on CDA heavily depend on prior knowledge. The stringent requirement of manual expertise may lead to unsatisfactory augmentation outcomes and impact the overall debiasing performance.

2.2.2. In-processing

In-processing methods change the language models themselves to achieve debiasing. An approach is to change the internal structure of language models and then retrain the models' parameters [37]. Stereotype Content Model (SCM) [38] is a theoretical framework for understanding stereotype content based on social psychology. SCM maps stereotype content into the psychological dimensions of warmth and competence to capture potential associations between bias and stereotypes, thereby helping to eliminate social bias. Dropout [10] adds an additional training procedure for BERT and ALBERT by increasing the attention weights and hiding the activation of the dropout parameters. Other approaches use the debiasing objective as a downstream task to fine-tune the parameters of language models. Context-Debias [39] fine-tunes the parameters of the PLMs to debias stereotypes by making them orthogonal to gender-related words. DebiasBERT [15] focuses on mitigating biases during BERT training, joint loss function and further pre-training of BERT to capture implicit biases in semantics, and reports performance on sentence completion and summary generation tasks. Auto-Debias [40] proposes a two-stage max-min debiasing method, which first maximizes the Jensen–Shannon divergence (JSD) between sensitive attribute words and stereotype words in different demographic groups to identify prompts and then minimizes the JSD to fine-tune the parameters of PLMs.

In-processing approach has some limitations. On the one hand, retraining based methods require expensive external resources and are quite time-consuming. On the other hand, although the fine-tuning based methods relatively reduce the cost, it is easy to introduce new biases when fine-tuning with abundant external corpora to destroy the language modeling capability.

2.2.3. Post-processing

Post-processing methods treat the language model as a black box to generate representations without changing the language model, and they perform a post-training process to remove biases before applying the representation to downstream tasks [41,42]. INLP trains a linear classifier to predict a target concept, such as gender, and then projects the representation into the null space of the classifier's weight matrix to remove bias from the representation [43]. Sent-Debias is a post-processing method for estimating the bias subspace of a sentence representation [44]. It augments the data by transforming bias attribute words into bias attribute sentences using a diverse set of sentence templates extracted from a text corpus. FairFil generates positive samples by replacing the original gender words with the opposite gender words and then applies a filter to contrastively debiasing the output of the sentence representation by the BERT encoder [45]. The sustainable debiasing method [46] is based on parametric-efficiency to mitigate potentially catastrophic forgetting during debiasing. It adds an adapter module after the encoding layer, and only updates the adapter parameters while freezing the PLMs parameters during the debiasing training process.

Post-processing method is an efficient debiasing method. It saves a lot of costs because it does not need to retrain the language model. Moreover, it does not change the PLMs' parameters thus not affecting the language modeling capabilities in downstream applications.

Our proposed CD^3 combines pre-processing and post-processing methods. Double data augmentation is a pre-processing method, which uses biased prompts from automatic search to perform a secondary data augmentation on the basis of CDA, so that data augmentation does not strictly depend on prior knowledge. Contrastive self-debiasing is a post-processing method that filters social biases in the sentence representation by inserting an adaptor after the PLMs' encoder. Note that during this process, we fix the parameters of the PLMs and train the adaptor only.

3. Preliminaries

Specific to demographic groups, a *social sensitive topic* $\mathcal{T} = \{T^1, T^2, \cdots, T^n, \cdots, T^N\}$ contains N *bias directions* T^n corresponding to each social subgroup [44]. Taking gender bias as an example, we denote social sensitive topic as $\mathcal{T} = \{\}\}Gender''\}$ with a binary

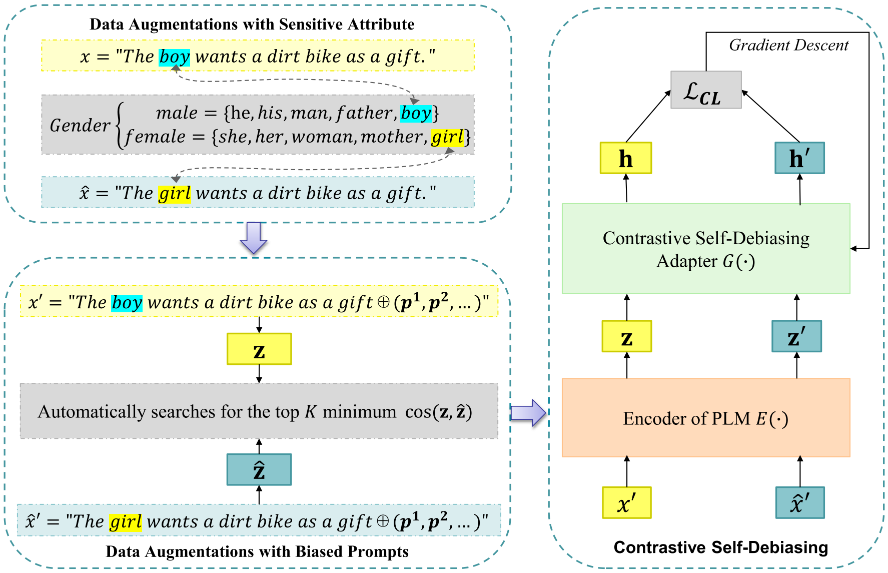
Data Augmentations with Sensitive Attribute
x = "The boy wants a dirt bike as a gift."
Gender
$male = \{he, his, man, father, boy\}$
$female = \{she, her, woman, mother, girl\}$
$\hat{x}$ = "The girl wants a dirt bike as a gift."
x' = "The boy wants a dirt bike as a gift $\oplus (\mathbf{p}^1, \mathbf{p}^2, \ldots)$"
$\mathbf{z}$
Automatically searches for the top K minimum $\cos(\mathbf{z}, \hat{\mathbf{z}})$
$\hat{\mathbf{z}}$
$\hat{x}'$ = "The girl wants a dirt bike as a gift $\oplus (\mathbf{p}^1, \mathbf{p}^2, \ldots)$"
Data Augmentations with Biased Prompts
Gradient Descent
$\mathcal{L}_{CL}$
$\mathbf{h}$
$\mathbf{h}'$
Contrastive Self-Debiasing Adapter $G(\cdot)$
$\mathbf{z}$
$\mathbf{z}'$
Encoder of PLM $E(\cdot)$
x'
$\hat{x}'$
Contrastive Self-Debiasing

Fig. 1. Overall architecture of CD^3. CD^3 is a two-stage model consisting of Double Data Augmentation stage and Contrastive Self-Debiasing stage. In the Double Data Augmentation stage, the original training corpora is processed by Data Augmentations With Sensitive Attribute and Data Augmentations With Biased Prompts successively to get the augmented training corpora. In the Contrastive Self-Debiasing stage, a contrastive self-debiasing adapter $G(\cdot)$ is trained to debias the social biases in sentence representation of the augmented training corpora.

bias direction $\{T^1, T^2\} = \{\}\}male'', \}\}female''\}$.[1] To augment the original training corpus, an M-tuple of *sensitive attribute word* $(t_1^n, t_2^n, \cdots, t_m^n, \cdots, t_M^n)$ is selected for each bias direction T^n from human stereotypes. Sensitive attribute words with the same position in different bias directions are the opposite words with the same semantics in different social subgroups. For gender bias with binary bias directions, examples of sensitive attribute words are as $(t_1^1, t_2^1, \cdots, t_5^1) = (\}\}he'', \}\}his'', \}\}man'', \}\}father'', \}\}boy'')$ and $(t_1^2, t_2^2, \cdots, t_5^2) = (\}\}she'', \}\}her'', \}\}woman'', \}\}mother'', \}\}girl'')$.

Let $\mathcal{M}$ denote a PLM and $E(\cdot)$ its encoder. Feeding the original corpus $\mathcal{X}$ containing sentences with social biases into $\mathcal{M}$ will obtain the embedding $\mathbf{z} = E(x)$ of sentence $x \in \mathcal{X}$. To mitigate the social biases in the embeddings encoded by PLMs, we match sentences in the training corpus using a set of sensitive attribute words $(t_1, t_2, \cdots, t_M)$ to obtain the training samples. Each training sample $x = (w^1, w^2, \cdots, w^L)$ is a sequence of words of length L and contains a sensitive attribute word t.

Our task objective is to train a self-debiasing adapter $f(\cdot)$ that mitigates the social biases implicit in the original sentence representation. The adapter $f(\cdot)$ takes the sentence embedding $\mathbf{z}$ output by the encoder $E(\cdot)$ of a PLM $\mathcal{M}$ as input and outputs the sentence representation $\mathbf{h} = f(\mathbf{z})$ after debiasing. It performs post-training process behind the encoder without updates PLMs' parameters.

4. Method

We propose a Contrastive Self-Debiasing model with Double Data augmentation (CD^3) to mitigate social biases specific to demographic groups in the PLMs' encoding. CD^3 consists of two stages: double data augmentation and contrastive self-debiasing. Algorithm 1 and Algorithm 2 describe the specific processes of double data augmentation and contrastive self-debiasing, respectively. We will elaborate on these two parts in the following subsections. The architecture of CD^3 is shown in Fig. 1.

4.1. *Double data augmentation*

In this stage, we perform a double data augmentation process on the training corpus to prepare for the next contrastive self-debiasing. Double data augmentation is achieved in two steps: 1) use sensitive attribute words to generate the positive sample pairs of the original training samples; 2) automatically search for texts that can maximize the bias between the positive sentence pairs as biased prompts to augment the positive sample pairs further.

4.1.1. *Data augmentation with sensitive attribute*

The first data augmentation replaces the sensitive attribute words in the original training samples with the corresponding sensitive attribute words in other bias directions.

[1] Following previous researches, we only considers a general binary gender subgroups, but in the real world it can be more diverse.

Specifically, for each sensitive attribute word t in any bias direction T of a social sensitive topic $\mathcal{T}$, we perform word matching on the original corpus and gather the sentences containing sensitive attribute words as the training corpus $\mathcal{X}$. Thus, each input sample $x = (w^1, w^2, \cdots, w^L) \in \mathcal{X}$ contains a sensitive attribute word denoted by w_t. Then, we replace the sensitive attribute word $w_t \in x$ with the corresponding sensitive attribute words $\{\hat{w}_t^1, \hat{w}_t^2, \cdots, \hat{w}_t^{N-1}\}$ in other bias directions $\{\hat{T}^1, \hat{T}^2, \cdots, \hat{T}^{N-1}\}$. For example, we set social sensitive topic $\mathcal{T} = \{\}\}Gender''\}$, bias direction $\{T^1, T^2\} = \{\}\}male'', \}\}female''\}$, sensitive attribute words $(t_1^1, t_2^1, \cdots, t_5^1) = (\}\}he'', \}\}his'', \}\}man'', \}\}father'', \}\}boy'')$ and $(t_1^2, t_2^2, \cdots, t_5^2) = (\}\}she'', \}\}her'', \}\}woman'', \}\}mother'', \}\}girl'')$. For the original input sentence $x =$*“The **boy** wants a dirt bike as a gift.”*, the sensitive attribute word $w_t =$*“boy”* corresponds to $\hat{w}_t =$*“girl”*. Then we replace *“boy”* with *“girl”*, and the positive samples sentence after the first data augmentation is $\hat{x} =$*“The **girl** wants a dirt bike as a gift.”*. The original sentence x and the augmented sentence $\hat{x}$ form a initial positive sample pair $(x, \hat{x})$, which are semantically consistent but represent opposite bias directions.

4.1.2. Data augmentation with biased prompts

Data augmentation with sensitive attribute words is a key part of debiasing, but it has some defects. On the one hand, pre-defined sensitive attribute words rely heavily on prior knowledge and consume artificial resources. On the other hand, simply replacing the sensitive attribute words may make the positive sample pairs too similar to each other, resulting in easy overfitting of the model. These defects are potential factors that affect the debiasing performance. Therefore, we seek an improved data augmentation method, which can carry out appropriate perturbations to the sample pairs so that the positive samples are not too similar and consume as little resources as possible.

Prompt learning [47] reformats NLP tasks to more directly elicit knowledge from models [48], which can guide PLMs to learn faster [49,50]. Most current works based on prompt use manual templates, for example in sentiment classification tasks, the most straightforward template like $f_{prompt(x)} = \}\}[X]\ Overall,\ it\ was\ a\ [Z]\ movie$”, where $[X]$ slot is inserted into the input and $[Z]$ slot represents the predicted target word. But the process of manually discovering the optimal prompts takes time and experience, which is a challenge for designers [51,52]. While the discrete prompts automatically induce for task-specific templates in the discrete space, eliminating manual design hassle.

Inspired by prompt learning, we propose a data augmentation method based on discrete prompts to perturb samples. The idea of prompt learning is to provide task-related prompts to guide the model to learn more directly. Our goal is to perturb the data, i.e. amplify the biased information between positive sample pairs. So we use the automatically searched biased text as task prompts, called *biased prompts*, which perturb by directly adding biased information to the sample. Therefore, after the first data augmentation, we perform a secondary data augmentation with biased prompts to obtain more biased positive sample pairs. The secondary data augmentation makes it difficult for the model to overfit in the debiasing training, so as to learn stronger debiasing ability and robustness.

Specifically, we automatically search for biased prompts in a given search space,[2] which are defined as texts that can elicit differences in representation between different demographic groups. The obtained biased prompts are aggregated into a set denoted as $\mathcal{P}$. Given a positive sample pair $(x, \hat{x})$ that outputs at the first data augmentation, we define the prompt template as $f_{prompt(x)} = [X] \oplus P$, where $[X]$ slot inserts the input sentence x or $\hat{x}$ and $P = (p^1, p^2, \cdots, p^D) \in \mathcal{P}$ represents the biased prompts consisting of D tokens p, $\oplus$ represents the string concatenation. The positive sample pair after concatenating the extra prompts is denoted as:

$$(x', \hat{x}') = (x \oplus (p^1, p^2, \cdots, p^D), \hat{x} \oplus (p^1, p^2, \cdots, p^D)). \tag{1}$$

The motivation to search biased prompts is to amplify the differences between positive sample pairs, maximizing the biases that language models exhibit toward different demographic groups. We approximate the differences between sample pairs by measuring the distance between two sentence embeddings using the Cosine Similarity. The Cosine Similarity among $(\mathbf{z}, \hat{\mathbf{z}})$ is denoted as $cos(\mathbf{z}, \hat{\mathbf{z}})$, where $\mathbf{z}$ and $\hat{\mathbf{z}}$ denote the embeddings of x' and $\hat{x}'$, respectively. For a given encoder $E(\cdot)$ of PLM $\mathcal{M}$, the embeddings $\mathbf{z} = E(x)$.

We aim to find the biased prompts that induce the lowest similarity between two sentence embeddings. There is a difference between the embeddings of samples in different bias directions. The biased prompts that cause the more significant difference indicate their stronger bias attributes, enabling the model to learn more robust debiasing performance. Because these texts can guide the model's learning, they can be called biased prompts.

We borrow a variant of beam search [15] as the search algorithm for biased prompts. Specifically, in the first iteration, we set the candidate space to the entire search vocabulary. For each input positive sample pair $(x, \hat{x})$, each candidate prompt P in the candidate space is inserted into the prompt template $f_{prompt(\cdot)}$ to generate $(x', \hat{x}')$. Then $(x', \hat{x}')$ is input into the encoder $E(\cdot)$ to obtain the sentence embeddings $(\mathbf{z}, \hat{\mathbf{z}})$. Next, we calculate the cosine similarity of all embeddings and sort the results from smallest to largest. Finally, we take the top K candidate prompts $\{P_1, P_2, \cdots, P_K\}$ as the search result of this iteration. The search results and the candidate prompts of the current round are then concatenated as the candidate space for the next iteration. The above process is repeated until the end of the iteration. The search results of each iteration constitute the final biased prompts set $\mathcal{P}$.

We describe in Algorithm 1 the overall process of double data augmentations. After applying double data augmentation to the original training corpus $\mathcal{X}$, we obtain the new training corpus $\mathcal{X}' = \{(x_1', \hat{x}_1'), (x_2', \hat{x}_2'), \cdots, (x_\eta', \hat{x}_\eta')\}$. We then feed the training corpus $\mathcal{X}'$ into the contrastive self-debiasing module.

[2] The search space can be any corpus including the training corpus. To avoid searching for non-text symbols, we use the top 5,000 most frequently occurring words in Wikipedia as the search vocabulary.

Algorithm 1 Double Data Augmentation.

```
Require: PLM's Encoder $E(\cdot)$, training corpus $\mathcal{X}$, sensitive attribute words $\{T, \hat{T}\}$, search space $\mathcal{V}$, beam width $K$, iteration number $\epsilon$;
Ensure: Augmentation corpus $\mathcal{X}'$;
// Step1. Data Augmentation With Sensitive Attribute
for $\forall x \in \mathcal{X}$ do
    Find out the word $w_t = (x = (w^1, w^2, \cdots, w^L)) \cap (T = (t_1, t_2, \cdots, t_M))$.
    Obtain $\hat{x}$ by replacing $w_t$ in $x$ with $\hat{w}_t \in \hat{T}$.
    Obtain the positive sample sentence pair $(x, \hat{x})$.
end for
// Step2. Data Augmentation With Biased Prompts
Set the candidate prompts $\mathcal{P}' = \mathcal{V}$
for 1 to $\epsilon$ do
    Obtain $(x', \hat{x}')$ by Eq. (1) for $\forall p \in \mathcal{P}'$.
    Calculate $cos(\mathbf{z}, \hat{\mathbf{z}})$ with $(\mathbf{z}, \hat{\mathbf{z}}) = (E(x'), E(\hat{x}'))$.
    Select the top $K$ candidate prompts $\mathcal{P}_{topk}$ with the smallest cosine similarity.
    Update $\mathcal{P}' = \{p \oplus v | \forall p \in \mathcal{P}_{topk}, \forall v \in \mathcal{V}\}$.
    Obtain the baised prompts set $\mathcal{P} = \mathcal{P} \cup \mathcal{P}_{topk}$.
end for
return $\mathcal{X}' = \{(x'_1, \hat{x}'_1), (x'_2, \hat{x}'_2), \cdots, (x'_\eta, \hat{x}'_\eta)\}$ by Eq. (1) for $\forall p \in \mathcal{P}$.
```

4.2. Contrastive self-debiasing

Contrastive learning is trained on unsupervised data and does not require specific downstream tasks. Its objective is to uniformize the representation of positive samples, which coincides with our debiasing objective. So we use a contrastive learning framework to train the debiasing model. To save on training costs, we apply a post-processing debiasing strategy, which is achieved by tuning an additional adapter and fixing PLMs' parameters. Therefore, we utilize a contrastive learning framework to learn a self-debiasing adapter in the contrastive self-debiasing stage. The self-debiasing adapter projects sentence embeddings from the original biased space to a new unbiased subspace, thereby removing the social biases implicit in the PLMs' encoding. It is a plug-and-play post-processing network, which does not need to change any internal structure and parameters of PLMs, and can be widely applied to the task of debiasing social biases for any PLM. The adapter is added behind the PLMs' encoder, and only the adapter parameters are tuned while the PLM parameters are frozen during debiasing training.

We train the model with the new corpus $\mathcal{X}'$ to mitigate social biases in the PLMs' encoding. Specifically, for each positive sample pair $(x'_i, \hat{x}'_i)_{1\le i\le \eta} \in \mathcal{X}'$, it is feed into the encoder $E(\cdot)$ to obtain the sentence embeddings as $(\mathbf{z}_i, \mathbf{z}'_i) = (E(x'_i), E(\hat{x}'_i))$. We then input $(\mathbf{z}_i, \mathbf{z}'_i)$ to the self-debiasing adapter $G(\cdot)$ to project it from the original embedding space to the new subspace, removing the social biases in the embeddings. The output of the adapter $G(\cdot)$ is denoted as $(\mathbf{h}_i, \mathbf{h}'_i) = (G(\mathbf{z}_i), G(\mathbf{z}'_i))$, which is the sentence representations after debiasing.

For a positive sample pair, we expect their representations to represent the same semantics in different bias directions, which means that the model treats different demographic groups equally on social sensitive topics. Therefore, our goal is to make the representations of positive sample pair $(\mathbf{h}_i, \mathbf{h}'_i)$ output by the self-debiasing adapter G as similar as possible by contrastive learning [53,54]. To learn the parameters of G, we utilize Mutual Information (MI) [55,56] as the contrastive loss function. MI is a general way to measure the degree of information among two variables and is defined as follows:

$$\mathcal{I}_{MI}(\mathbf{x}, \mathbf{y}) := \mathbb{E}_{p(\mathbf{x},\mathbf{y})}[\log \frac{p(\mathbf{x}, \mathbf{y})}{p(\mathbf{x})p(\mathbf{y})}], \tag{2}$$

where $p(\mathbf{x})$ and $p(\mathbf{y})$ are the marginal distribution probability of variable $\mathbf{x}$ and variable $\mathbf{y}$, $p(\mathbf{x}, \mathbf{y})$ is the joint distribution of two variables $(\mathbf{x}, \mathbf{y})$. By maximizing MI among the two sentence representations $(\mathbf{h}_i, \mathbf{h}'_i)$, we can reduce the semantic distance of different bias directions. Under the goal of MI, the adapter is trained to debias using the self-information of the text. Since maximizing MI is challenging in practice, we minimize the lower bound of $\mathcal{I}_{\mathcal{MI}}$ based on the Noise Contrast Estimation (NCE) [57] defined as follows:

$$\mathcal{I}_{NCE} := \frac{1}{N} \sum_{i=1}^{N} \log \frac{exp(f(\mathbf{x}_i, \mathbf{y}_i))}{\frac{1}{N} \sum_{j=1}^{N} exp(f(\mathbf{x}_i, \mathbf{y}_j))}, \tag{3}$$

where $f(\cdot, \cdot)$ is a scoring function. Thus, our loss function is denoted as:

$$\mathcal{L}_{CL} = -\frac{1}{\eta} \sum_{i=1}^{\eta} \log \frac{exp(cos(\mathbf{h}_i, \mathbf{h}'_i)/\tau)}{\frac{1}{\eta} \sum_{j=1}^{\eta} exp(cos(\mathbf{h}_i, \mathbf{h}_j)/\tau)}, \tag{4}$$

where η is the sample pairs number in a batch, τ is the temperature parameter, $(\mathbf{h}_i, \mathbf{h}'_i)$ is the representations of a positive sample pair, and $(\mathbf{h}_i, \mathbf{h}_j)_{j\ne i}$ is the representations of a negative sample pair obtained by combining the original sample and the other samples in a batch. By maximizing the loss function, the distance between the positive sample pairs is narrowed and the distance between the negative sample pairs is widened. We minimize the loss function $\mathcal{L}_{CL}$ and apply gradient descent to learn parameters of the adapter G. We describe the training process of contrastive self-debiasing in Algorithm 2.

Algorithm 2 Contrastive Self-Debiasing.

```
Require: PLM's Encoder $E(\cdot)$, augmentation corpus $\mathcal{X}'$, batch size $\eta$;
Ensure: Self-debiasing adapter $G(\cdot)$;
for each batch do
    Encoding each positive sample pair $(x_i', \hat{x}_i')_{1\le i\le \eta} \in \mathcal{X}'$ into $(\mathbf{z}_i, \mathbf{z}_i') = (E(x_i'), E(\hat{x}_i'))$.
    Obtain the representations $(\mathbf{h}_i, \mathbf{h}_i') = (G(\mathbf{z}_i), G(\mathbf{z}_i'))$.
    Calculate $\mathcal{L}_{CL}$ with $(\mathbf{h}_i, \mathbf{h}_i')$ by Eq. (4).
    Update the parameters of $G$ by minimizing $\mathcal{L}_{CL}$ and gradient descent.
end for
return $G(\cdot)$.
```

5. Experiments

We conduct extensive experiments to demonstrate the effectiveness of our proposed debiasing model CD^3. Concretely, we investigate the following three research questions:

Q1. Compared with state-of-the-art methods, how effective is CD^3 in mitigating social biases of PLMs' encoding?

Q2. How does the double data augmentations affect debiasing performance of CD^3?

Q3. Does CD^3 have an impact on the language modeling capabilities of PLMs on downstream tasks?

5.1. Experimental setup

We first elaborate on the experimental setup, including the sensitive attribute words, the training datasets, the baseline debiasing models, the evaluation metric of social biases, and the implementation details of experiments.

5.1.1. Sensitive attribute words

In the experiments, we take gender bias [28] and race bias [29] as case studies and set social sensitive topics as {*Gender*, *Race*}. For *Gender*, we set the binary bias directions as (*Male*, *Female*) and pre-define the sensitive attribute words with reference to Sent-Debias and FairFil as follows:

(*Male*, *Female*) = {(*man*, *woman*), (*boy*, *girl*), (*he*, *she*), (*father*, *mother*), (*son*, *daughter*), (*guy*, *gal*), (*male*, *female*), (*his*, *her*), (*himself*, *herself*), (*John*, *Mary*)}.

For *Race*, much previous work is limited to ambiguous sensitive attribute words, such as *black* and *white*, which are highly likely to represent color as well as race in the samples. Therefore, in order to clearly distinguish between races, we mainly define sensitive attribute words based on the geographical location of different countries. With reference to StereoSet [58], we select some of the top race-related stereotype words from the StereoSet based on artificial experience and extend them appropriately. We set the binary bias directions as (*European-American*, *African-American*) and pre-define the sensitive attribute words as follows:

(*European-American*, *African-American*) = {(*european*, *african*), (*british*, *african*), (*german*, *african*), (*polish*, *african*), (*russian*, *african*), (*europe*, *african*), (*italian*, *african*), (*portuguese*, *african*), (*french*, *african*), (*romanian*, *african*), (*greek*, *african*), (*irish*, *african*), (*spanish*, *african*), (*bosnian*, *african*), (*albanian*, *african*), (*caucasian*, *african*), (*caucasian*, *nigerian*), (*caucasian*, *ethiopian*), (*caucasian*, *africa*), (*caucasian*, *ghanaian*), (*caucasian*, *kenyan*), (*caucasian*, *mexican*), (*caucasian*, *somali*), (*caucasian*, *liberian*), (*caucasian*, *moroccan*), (*caucasian*, *cameroonian*), (*caucasian*, *south-african*), (*caucasian*, *eritrean*), (*caucasian*, *sudanese*), (*caucasian*, *egyptian*)}.

5.1.2. Datasets

In the double data augmentation stage, we aim to identify more representative prompts (that is, texts with biased attributes) and then expect the training corpus to match the social biases in demographic groups as much as possible. In the contrastive self-debiasing stage, our training objective is to train a adapter using augmented samples, so the texts derived from the real world are more appropriate. Therefore, we use different training datasets in the double data augmentation and the contrastive self-debiasing to get closer to their learning objectives.

In the double data augmentation stage, for the social sensitive topics *Gender*, we use 269 gender-related corpora from CrowS-Pairs [26] as input texts. For the social sensitive topics *Race*, since the race-related corpora in CrowS-Pairs does not match our pre-defined sensitive attribute words, we randomly screen 1,086 race-related corpora from the toxic language from Twitter[3] based on sensitive attribute words as the input texts. We apply the double data augmentation on the gender-related corpora and the race-related corpora, respectively, and then identify the prompts with strong biased attributes in order to strengthen the self-debiasing effect.

In the contrastive self-debiasing stage, for the social sensitive topic *Gender*, we follow previous work [44,45] by matching training data from five real-world datasets: POM [59], MELD [60], WikiText-2 [61], Reddit [62] and Stanford Sentiment Treebank [63]. These five datasets contain more than 180,000 instances, and about 20,438 instances are obtained by matching gender-related sensitive attribute words. For the social sensitive topic *Race*, we match a corpus of 13,839 instances from the toxic language from Twitter as training data.

[3] https://www.kaggle.com/datasets/fizzbuzz/cleaned-toxic-comments.

5.1.3. Baselines

We consider the following state-of-the-art debiasing methods as baseline models.

- **CDA** [10] is a debiasing technique applied to databases, which generates counterfactual examples by exchanging sensitive attribute words in the database to balance the corpus.
- **Dropout** [10] adds an additional pre-training process for debiasing by applying a dropout regularization for attention weight and hidden activations to BERT and ALBERT.
- **Sent-Debias** [44] improves the word embedding debiasing method [7] to the sentence embedding debiasing method, which uses principal component analysis to estimate the variation direction of the representation set projected into the bias subspace.
- **INLP** [43] trains a linear classifier and then debias the representation by projecting the representation into the nullspace of the classifier's weight matrix.
- **FairFil** [45] improves the sentence-level debiasing method of Sent-Debias by introducing a contrastive learning framework, which trains a fairness filter to debiasing the output of the pre-trained encoder.
- **Auto-Debias** [40] is an in-processing method that automatically mitigates biases through distribution alignment loss and exploits prompts from cloze-style completions to explore the biases encoded by PLMs.

5.1.4. Evaluation metrics of social biases

We select two intrinsic bias evaluation metrics, SEAT [25] and CrowS-Pairs [26], and one extrinsic bias evaluation metric, Unintended Bias Metrics [27], to evaluate the effectiveness of our proposed debiasing model and all baseline models in mitigating the social biases of PLMs. These evaluation metrics can measure a variety of social biases in demographic groups, such as gender bias and race bias, and are widely used in fairness research.

Sentence Encoder Association Test (SEAT) [25] extends the Word Embedding Association Test (WEAT) [8] to sentence-level representation. WEAT evaluates the biases of word embeddings by measuring the association between two groups of attribute words (e.g., *man* and *woman*) and two groups of stereotyping target words (e.g., *family* and *career*). Formally, the sets of attribute words are denoted by $\mathcal{A}$ and $\mathcal{B}$, and the sets of target words are denoted by $\mathcal{X}$ and $\mathcal{Y}$. Then the SEAT test statistics are defined as follows:

$$s(\mathcal{X},\mathcal{Y},\mathcal{A},\mathcal{B}) = \sum_{x\in\mathcal{X}} s(x,\mathcal{A},\mathcal{B}) - \sum_{y\in\mathcal{Y}} s(y,\mathcal{A},\mathcal{B}), \tag{5}$$

where $s(w,\mathcal{A},\mathcal{B})$ represents the difference between the average of the cosine similarity of word w with all words in $\mathcal{A}$ and the average of the cosine similarity of word w to all words in $\mathcal{B}$, and it is defined as follows:

$$s(w,\mathcal{A},\mathcal{B}) = \frac{1}{|\mathcal{A}|}\sum_{a\in\mathcal{A}} cos(w,a) - \frac{1}{|\mathcal{B}|}\sum_{b\in\mathcal{B}} cos(w,b), \tag{6}$$

where $w \in \mathcal{X}$ or $\mathcal{Y}$, and $cos(\cdot,\cdot)$ represents the cosine similarity. They report the normalized effect size as follows:

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

where $\mu(\cdot)$ is the mean function and $\sigma(\cdot)$ is the standard deviation.

SEAT uses simple templates such as "*This is a [word]*" to substitute attribute words and target words to obtain context-independent embeddings. Then the SEAT test statistic is computed similar to Eq. (7) for sentence-level representations. Ideally, we expect the PLMs to be fair to different demographic groups, that is, different attribute words and stereotype target words are equidistant from each other. We report the effect size of SEAT in the experimental results, and the closer it is to 0, the more fair the embedding is. We use SEAT 6, 6b, 7, 7b, 8, and 8b to estimate gender bias and SEAT 3, 3b, 4, 5, and 5b to estimate race bias. We also report the average of all subtests' absolute values as "Avg.esize".

Crowdsourced Stereotype Pairs (CrowS-Pairs) [26] is a crowdsourced dataset containing sentence pairs. Each sentence pair is semantically similar and differ only in a few tokens. The first sentence represents the stereotype of the disadvantaged group and the second sentence is the version that violates the stereotype, such as "*Women don't know how to drive*" and "*Men know how to drive*".

Some recent work [64,65] suggests that the method of the pseudo-likelihood-based scoring [66] originally used by CrowS-Pairs may be affected by model calibration issues. Thus, following previous work [67], we adopt masked token probabilities similar to StereoSet [58] for computing template sentence scores. We define the stereotype score of a model as the percentage of examples for which the model assigns a higher mask token probability to the stereotyped sentence. We report the CrowS-Pairs stereotype scores on gender and race in the experimental results, where its ideal value is 50%.

Unintended Bias Metrics (UBM) [27] is an extrinsic bias measure that uses toxic text classification tasks to assess the extent of a model's social bias. It creates a large test set of 1.8 million samples from online reviews, each of which is manually labeled for toxicity. The dataset contains 450,000 samples of the identity groups to which they are tagged. The fairness of PLM is represented by the variation in the distribution of classifiers between different identity groups. Specifically, UBM consists of a suite of five metrics, including three metrics based on the Area Under the Receiver Operating Characteristic Curve (AUC), **Subgroup AUC**, **BPSN AUC**, and **BNSP AUC**, as well as two Average Equality Gap (AEGs), **Negative AEG** and **Positive AEG**. AUC-based metrics are calculated based on the negative/positive mis-ordering between identity subgroups and backgrounds, and they are defined as follows:

$$\text{Subgroup AUC} = \text{AUC}(D_g^- + D_g^+), \tag{8}$$

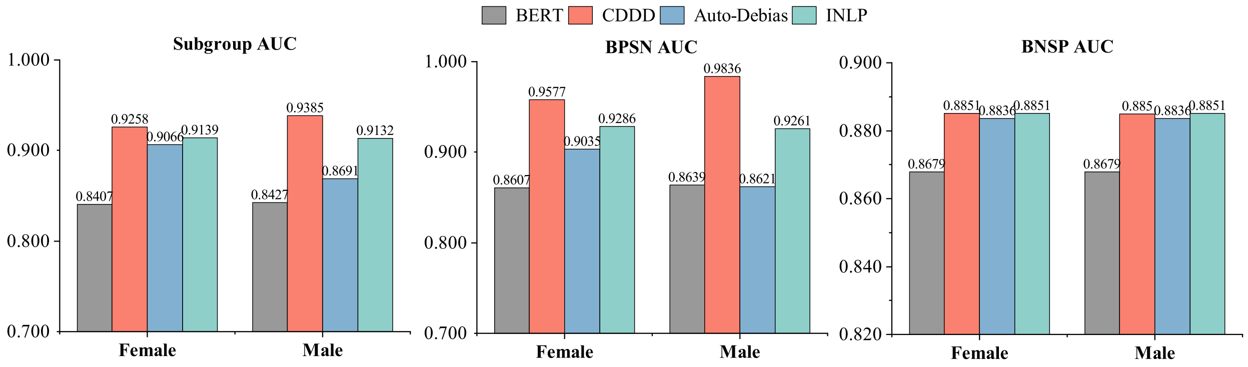

Fig. 2. Gender subgroup results of AUC-based metrics in the UBM experiments. The closer the score is to 1, the less biased the model is to the subgroup.

$$\text{BPSN AUC} = \text{AUC}(D^+ + D_g^-), \tag{9}$$

$$\text{BNSP AUC} = \text{AUC}(D^- + D_g^+), \tag{10}$$

where D^+ and D^- represent the positive and negative examples of the background set, D_g^+ and D_g^- represent the positive and negative examples of the identity subgroup. Subgroup AUC reflects the model's understanding and separability of target subgroups. BPSN AUC represents the Background Positive Subgroup Negative AUC, which decreases when the negative example score of the subgroup is higher than the positive example score of the background, similar to the false positive scenario. BNSP AUC represents the Background Negative Subgroup Positive AUC, which decreases when the positive example score of the subgroup is lower than the negative example score of the background, similar to the false negative scenario. These three AUC-based metrics can be combined to measure the degree to which the model is biased against the target subgroup. The closer their score is to 1, the less biased the model.

AEGs builds on the Equality Gap metric, calculating the difference in true positive rates between subgroups and backgrounds under a specific threshold. It describes more subtle differences in distribution that are difficult to observe on the AUC-based metrics. Positive AEG and Negative AEG are defined as follows:

$$\text{Positive AEG} = \frac{1}{2} - \frac{\mathrm{MWU}(\mathrm{D}_g^+, \mathrm{D}^+)}{|D_g^+||D^+|}, \tag{11}$$

$$\text{Negative AEG} = \frac{1}{2} - \frac{\mathrm{MWU}(\mathrm{D}_g^-, \mathrm{D}^-)}{|D_g^-||D^-|}, \tag{12}$$

where $\mathrm{MWU}(\cdot,\cdot)$ is the Mann-Whitney U test statistic [68]. A score closer to 0 for AEGs indicates less social bias.

5.1.5. Implementation details

We consider three more general PLMs BERT [16], ALBERT [17], and RoBERTa [18]. All experiments use the checkpoints as *bert-base-uncased*, *albert-base-v2*, and *roberta-base* implemented based on Huggingface Transformers library [69].

In the double data augmentation stage, we take the top 5,000 most frequently occurring words in Wikipedia[4] as the search space $\mathcal{V}$ for biased prompts, set the iteration number ϵ to 5, and set the beam width K to $\{10, 20\}$ that will result in $\{50, 100\}$ biased prompts after searching. Specifically, we report the results of CD^3 with $K = 20$ in the debiasing performance experiments (Table 1 and Table 2) and contrast the results with $K = 10$ in the ablation experiments (Table 3 and Table 4).

In the contrastive self-debiasing stage, we train the self-debiasing adapter $G(\cdot)$ as the one-layer MLP with the ReLU activation function [45]. We set the learning rate to 1×10^{-5}, the batch size η to 128, and the epoch to 50, the temperature parameter τ to 1. In debiasing training, we only tune the parameters of the self-debiasing adapter and fix the parameters of PLMs. For each experiment, we report the average results of five runs.

In the SEAT experiment, sentence embeddings are calculated by averaging the embeddings of all tokens in the sentence [67]. In the UBM experiment, we use subgroups "*Male*" and "*Female*" to measure gender bias, and subgroups "*White*" and "*Black*" to measure race bias. We use the comment dataset created by UBM to train and test the models, dividing the training set and the test set in a 9:1 ratio. In the training phase, the batch size is set to 256, the learning rate to 1×10^{-5}, and the epoch to 10. Samples with toxicity greater than 0.5 are considered toxic (positive example), and the threshold for AEGs is set to 0.5 [27].

To ensure fairness in comparison, we rerun the experimental results for all baseline models, keeping our experimental settings as consistent as possible. CDA uses the sensitive attribute words of *Gender* and *Race* as same as CD^3. All baseline models that require a training dataset use the same training datasets as CD^3. Experimental results for CDA, Dropout, Sent-Debias, and INLP are obtained by rerunning on the code provided by the literature [67]. The results for FairFil and Auto-Debias are obtained from models retrained with the source code they provided.

[4] https://github.com/IlyaSemenov/wikipedia-word-frequency.

Table 1
Gender debiasing results of SEAT and CrowSPairs on BERT, ALBERT, and RoBERTa. The best result is indicated in **bold**.

Metric Model	SEAT-6	SEAT-6b	SEAT-7	SEAT-7b	SEAT-8	SEAT-8b	Avg.esize	Stereo	Anti-Stereo	Overall
BERT	0.9315	0.0895	-0.1240	0.9366	0.7827	0.8584	0.6205	57.86	56.31	57.25
+CDA	0.5960	-0.1030	-0.2362	0.8003	0.3941	0.7338	0.4772	55.35	56.86	55.73
+Dropout	0.9116	0.1208	0.3212	0.8573	0.7766	0.8670	0.6424	55.35	60.19	57.25
+Sent-Debias	**0.3356**	-0.3143	-0.6237	**0.5140**	0.3910	0.4356	0.4357	42.14	70.87	53.44
+FairFil	0.6825	-0.1397	-0.6158	0.8393	**0.0489**	-0.5007	0.4711	62.89	**49.51**	57.63
+Auto-Debias	0.3733	-0.0560	0.7450	1.1748	0.8561	0.8233	0.6714	52.83	56.31	54.2
+INLP	0.6191	-0.2264	0.3263	0.5911	0.4295	0.5493	0.4570	**50.94**	62.14	55.34
+CD^3 (ours)	0.6670	**0.0298**	**-0.0187**	0.5807	-0.2790	**-0.2935**	**0.3115**	54.09	51.46	**53.05**
ALBERT	0.6368	0.1507	0.4872	0.9559	0.6832	0.8233	0.6229	47.80	**48.54**	48.09
+CDA	0.6374	-0.1480	0.4584	1.0557	0.7825	1.2221	0.7173	53.46	**51.46**	52.67
+Dropout	0.6805	-0.3155	0.2412	1.2233	0.7610	1.2829	0.7507	51.90	61.17	55.34
+Sent-Debias	0.4326	-0.0990	-0.2847	0.6995	0.5769	0.7699	0.4771	25.16	82.52	47.71
+FairFil	-0.6280	**-0.0447**	0.4529	0.9387	0.4034	0.7793	0.5412	31.45	77.67	49.62
+Auto-Debias	0.2821	-0.2240	-0.6112	1.2660	-0.1272	1.0760	0.5978	57.23	38.83	**50.00**
+INLP	0.4875	-0.1286	-0.3743	0.7776	0.2829	0.7753	0.4710	33.96	70.87	48.47
+CD^3 (ours)	**0.1253**	-0.1707	**-0.0195**	**-0.0087**	**0.0233**	**0.0636**	**0.0685**	**50.31**	53.40	51.53
RoBERTa	0.9222	0.2076	0.9788	1.4597	0.8103	1.2608	0.9399	67.92	**48.04**	60.15
+CDA	0.8405	**-0.0351**	0.3947	0.8570	0.7082	0.7245	0.5933	54.76	53.49	54.65
+Dropout	0.9514	0.0375	0.3681	0.8687	**0.3902**	0.8323	0.5747	62.26	42.16	54.41
+Sent-Debias	0.9671	-0.0498	0.4319	1.3827	0.5584	1.2576	0.7746	45.91	60.78	51.72
+FairFil	0.4714	0.1513	0.7202	0.7057	0.8397	0.8930	0.6302	**46.20**	44.12	45.21
+Auto-Debias	0.7671	0.1898	0.8240	1.1169	0.5436	0.9006	0.7237	38.99	53.92	44.83
+INLP	0.7363	0.0470	0.8362	1.3530	0.7434	1.2141	0.8217	35.22	58.82	44.44
+CD^3 (ours)	**0.4571**	-0.2058	**-0.3610**	**0.6628**	0.8038	**0.7153**	**0.5343**	45.28	52.94	**48.28**

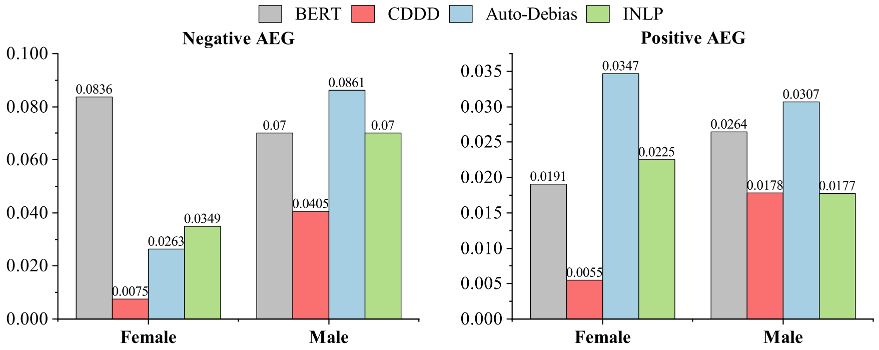

Fig. 3. Gender subgroup results of AEG metrics in the UBM experiments. The closer the score is to 0, the less biased the model is to the subgroup.

5.2. Debiasing performance

To demonstrate the effectiveness of CD^3, we conduct abundant experiments of gender debiasing and race debiasing on BERT, ALBERT, and RoBERTa. The results of intrinsic bias metrics in SEAT and CrowS-Pairs are reported in Table 1 and Table 2. We report the original bias measurement results of BERT, ALBERT, and RoBERTa for the readers' reference. For the SAET, we report the effect size for each subtest and the average of all subtests' absolute values, with values close to 0 indicating better results. For the Crows-Pairs, we report the percentage by which the model tends to stereotype and anti-stereotype, as well as the overall score, with closer to 50% indicating that the model is less biased. In addition, we evaluate the model's extrinsic bias in the toxic text classification task. The experimental results of the gender subgroup are shown in Fig. 2 and Fig. 3, and the experimental results of the race subgroup are shown in Fig. 4 and Fig. 5. We use BERT as the base model and compare CD with the original BERT, Auto-Debias and INLP. We report the five metrics of USM, where the ideal values of Subgroup AUC, BPSN AUC, and BNSP AUC are 1, and the ideal values of Positive AEG and Negative AEG are 0.

5.2.1. Gender debiasing

From the SEAT test results in Table 1, our model effectively removes the gender bias in BERT, ALBERT, and RoBERTa in most subtest scenarios. We reduce the average effect size for BERT from 0.6205 to 0.3115, for ALBERT from 0.6229 to 0.0685, and for RoBERTa from 0.9399 to 0.5343. Compared with other debiasing methods, CD^3 achieves better debiasing performance in general. On BERT, CD^3 achieves optimal results on SEAT 6b, 7, 8b and average effect size, and sub-optimal results on SEAT 7b and 8. On ALBERT, CD^3 achieves the most significant debiasing effect on all subtests except SEAT 6b. Compared with BERT and ALBERT, RoBERTa has the highest original bias, and the debiasing effect of all models is not obvious. Despite this, CD^3 still achieves the

Table 2
Race debiasing results of SEAT and CrowSPairs on BERT, ALBERT, and RoBERTa. The best result is indicated in **bold**.

Metric Model	SEAT-3	SEAT-3b	SEAT-4	SEAT-5	SEAT-5b	Avg.esize	Stereo	Anti-Stereo	Overall
BERT	0.7777	0.4688	0.9007	0.8870	0.5386	0.7146	64.41	39.53	62.33
+CDA	0.7819	0.2014	0.9003	0.9341	0.4061	0.6448	64.41	37.21	62.14
+Dropout	0.7938	0.3256	0.8821	0.9222	0.4603	0.6768	62.92	39.53	60.97
+Sent-Debias	0.7774	0.4714	0.9004	0.8855	0.5173	0.7104	64.62	41.86	62.72
+FairFil	-0.2461	**-0.0499**	-0.4062	-0.3587	0.2599	0.2642	27.54	60.47	30.29
+Auto-Debias	0.7169	0.4812	0.7937	0.9029	0.4985	0.6786	67.37	39.53	65.05
+INLP	0.7928	0.2966	0.8945	0.9088	0.4295	0.6644	62.50	41.86	60.78
+CD^3 (ours)	**-0.0657**	-0.1280	**-0.3783**	**-0.3549**	**-0.1291**	**0.2112**	**41.53**	**51.16**	**42.33**
ALBERT	1.1319	-0.2519	0.9558	1.0412	0.0578	0.6877	64.62	39.53	62.52
+CDA	1.1130	-0.0702	1.0129	1.2937	0.2530	0.7486	58.05	46.51	57.09
+Dropout	0.7972	-0.0839	0.8287	1.2032	0.5783	0.6983	61.44	53.49	60.78
+Sent-Debias	1.1305	-0.2154	0.9564	1.0445	0.0941	0.6882	65.68	41.86	63.69
+FairFil	1.0523	-0.1712	0.8620	0.9031	0.0961	0.6169	55.93	**51.16**	55.53
+Auto-Debias	1.1569	-0.3503	1.0233	1.0526	0.1438	0.7454	66.53	41.86	64.47
+INLP	1.1265	-0.3241	0.9084	1.0931	-0.1246	0.7153	59.75	46.51	58.64
+CD^3 (ours)	**0.5005**	**-0.0136**	**0.0128**	**-0.0256**	**0.0023**	**0.1110**	**54.24**	**48.84**	**53.79**
RoBERTa	-0.1139	-0.0026	-0.3146	0.7804	0.3862	0.3195	64.83	51.16	63.57
+CDA	-0.1206	**0.0002**	-0.3214	0.5287	0.0513	0.2004	44.61	48.84	44.96
+Dropout	-0.5447	-0.0443	-0.7675	0.5756	0.3133	0.4491	62.37	46.51	61.05
+Sent-Debias	0.3545	0.1298	0.2251	0.5363	0.3013	0.3094	68.86	46.51	66.86
+FairFil	-0.2357	-0.0168	-0.3056	0.3705	0.2615	0.2380	36.79	53.49	38.18
+Auto-Debias	-0.2617	-0.0872	-0.5042	0.6420	0.3496	0.3689	59.62	48.84	58.72
+INLP	-0.0749	0.0677	-0.2555	0.7250	0.2766	0.2799	62.58	53.49	61.82
+CD^3 (ours)	**-0.0637**	0.0568	**-0.1924**	**0.3337**	**0.0479**	**0.1389**	**51.57**	**50.98**	**51.34**

lowest average effect size score and performs best across multiple subgroups. In addition, our method has a more stable performance than the baseline methods.

From the results of CrowS-Pairs in Table 1, our debiasing model is relatively fair for stereotyping and anti-stereotyping. For the original BERT, ALBERT, and RoBERTa, CD^3 decreases their CrowS-Pairs stereotype scores. CD^3 obtains the most outstanding overall scores on both BERT and RoBERTa, and while the overall score on ALBERT is not the best, our combined performance on all three scores exceeds the baseline models. It can be observed that the overall performance of CD^3 on CrowS-Pairs is not as good as on SEAT. We analyze the reason is that CD^3 is a debiasing method based on sentence-level representation, while CrowS-Pairs is measured by masked for each token. Encouragly, our results still exhibit an acceptable debiasing performance under this disadvantage.

From the results of UBM in Figs. 2 and 3, CD^3 shows the strongest debiasing performance in the toxic text classification task on BERT. On the Subgroup AUC, all three debiasing models improve the original BERT score, and CD^3 improves more significantly than Auto-Debias and INLP. This suggests that CD^3 promotes BERT's understanding of gender subgroups rather than merely alleviating bias. On the BPSN AUC, CD^3 increases the scores of the female subgroup and the male subgroup to 0.9577 and 0.9836, indicating a remission of false positives in the gender samples. The BNSP AUC of the four models are slightly different, indicating that the debiasing methods are not significant in alleviating the false negative situation in the gender subgroups. Encouragingly, CD^3 is still performing well. The three AUC-based metrics in Fig. 2 confirm that applying CD^3 to BERT not only improves the ability to understand gender subgroups, but also mitigates the mis-ordering of false positive and false negative samples. The two AEG metrics in Fig. 3 can describe more subtle distribution shifts that the AUC-based metrics cannot capture. In both gender subgroups, CD^3 achieves the lowest score for both Negative AEG and Positive AEG, demonstrating the best separability of our proposed model in the gender subgroups. The results of UBM experiments prove that CD^3 has excellent performance in gender debiasing in toxic text classification tasks.

5.2.2. Race debiasing

According to the SEAT test results in Table 2, the effectiveness of our proposed method on race debiasing is significant. CD^3 reduces the average effect size of the original BERT from 0.7146 to 0.2112, the original ALBERT from 0.6877 to 0.1110, and the original RoBERTa from 0.3195 to 0.1389. Our model achieves the best debiasing effect in almost all SEAT subtests. Our experimental results are more balanced and stable compared to the baseline models, which perform unstable across different subtests. We find negligible improvement in average effect size of CDA, Dropout, Sent-Debias, Auto-Debias, INLP on BERT, as most of them do not consider race bias. It is worth noting that FairFil retrained with CD^3's sensitive attribute words about race gives competitive results in effect size, which verifies the effectiveness of our designed sensitive attribute words. Moreover, on ALBERT, the performance of all baselines is quite unsatisfactory. Compared with BERT and ALBERT, the original RoBERTa is less racially biased, which also leads to the insignificant or even reverse debiasing effect of baseline models on RoBERTa. Surprisingly, CD^3 exhibits a prominent debiasing advantage on both ALBERT and RoBERTa.

According to the CrowS-Pairs results in Table 2, CD^3 obtains improved stereotype scores for all BERT, ALBERT, and RoBERTa on race debiasing. Relative to the other debiasing models, CD^3 achieves nearly 50% stereotype and anti-stereotype scores in three PLMs, indicating that it has higher fairness. We note that Fairfil shows a pronounced bias for BERT on CrowS-Pairs different from its

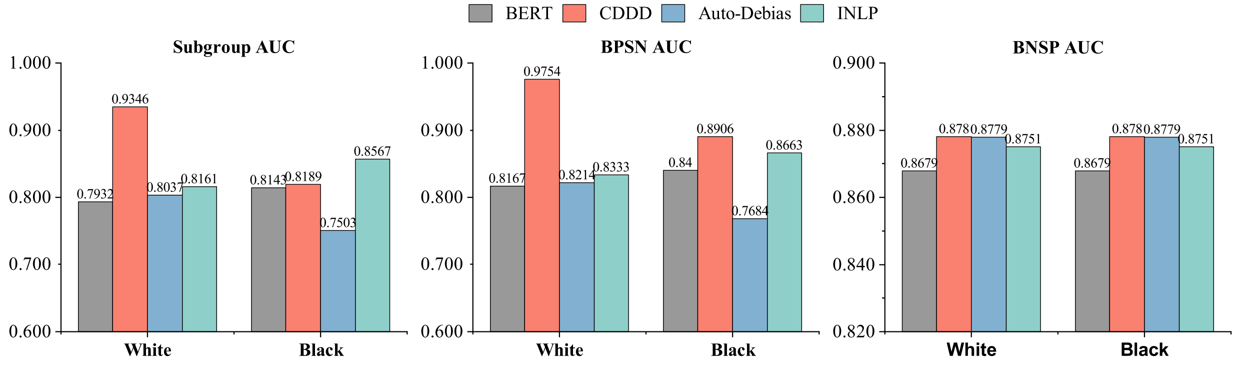

Fig. 4. Race subgroup results of AUC-based metrics in the UBM experiments. The closer the score is to 1, the less biased the model is to the subgroup.

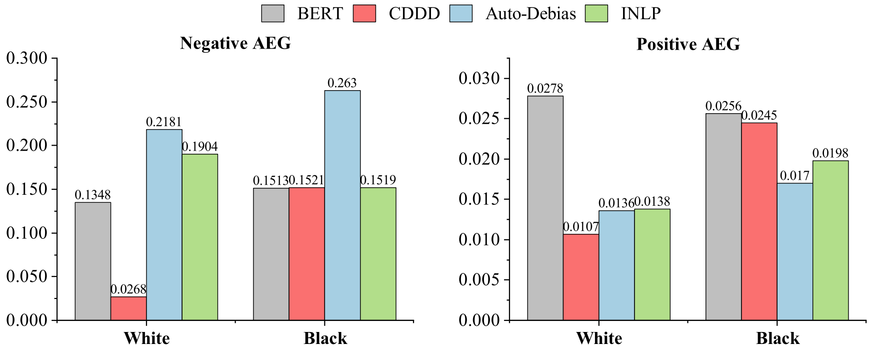

Fig. 5. Race subgroup results of AEG metrics in the UBM experiments. The closer the score is to 0, the less biased the model is to the subgroup.

fair behavior on the SEAT test. Referring to some work [70,71], they demonstrate that CrowS-Pairs exhibits unstable performance with different pre-training runs on PLMs. Therefore, we recommend that readers and researchers do not pay too much attention to the experimental results of CrowS-Pairs and try to observe from multiple metrics.

According to the results of UBM in Figs. 4 and 5, all models' performance on race subgroups is similar to that on gender subgroups. And consistently, compared to the baseline models, the debiasing performance of CD^3 is the most impressive overall. From the results of Subgroup AUC, BPSN AUC, and BNSP AUC in Fig. 4, CD^3 improves the original BERT's understanding of race subgroups and corrects the mis-ordering of false positive and false negative samples. The Negative AEG and Positive AEG metrics in Fig. 5 reveal the score distribution within the race subgroups for each model. The Negative AEG score of Auto-Debias indicates that it causes BERT to produce a larger distribution shift, which is contrary to its main purpose of debiasing. While CD^3 reduces the original distribution shift, demonstrating the validity of our proposed model. Compared with the Black subgroup, the White subgroup achieves a more obvious debiasing effect, which is reflected in that the score of Subgroup AUC increases from 0.7932 to 0.9346, and the score of BPSN AUC increases from 0.8167 to 0.9754, as well as a significant decrease in both AEG metrics. This shows that the debiasing model has different debiasing degrees for different subgroups.

We observe that most baseline models perform poorly in terms of race debiasing. This is a challenge for current language model fairness research, that is, most methods only focus on gender debiasing and they struggle to generalize to other social biases beyond gender. One of the reasons is that existing sensitive attribute words about race are limited or challenging to cover comprehensively, which is crucial for screening the training corpus. Encouragingly, in ALBERT's experiments with gender debiasing and race debiasing, our model shows near-fair performance, with effect sizes close to 0 for most subtests. The sensitive attribute words designed by us have a certain effect on helping to mitigate bias. However, designers need to explore further to fully cover race bias in society. In our proposed CD^3 model, the double data augmentation strategy alleviates the dependence of data augmentation on human experience to some extent by using biased prompts to strengthen the model's attention to social biases.

In summary, from the two aspects of intrinsic bias and extrinsic bias, our proposed contrastive self-debiasing model with double data augmentation has excellent performance of gender debiasing and race debiasing on BERT, ALBERT, and RoBERTa, while having relatively stable generalization performance. We have answered the first question (**Q1**) proposed at the beginning of this section.

5.3. Ablation studies

In order to explore the influence of the double data augmentation on CD^3 debiasing performance, we conduct ablation studies on BERT and ALBERT. We compare the effect of biased prompts on debiasing performance in two dimensions: quantity and probing method. We report the effect sizes of SEAT test statistics for gender debiasing and race debiasing in Table 3 and Table 4, respectively.

Table 3
Gender debiasing results of SEAT on BERT and ALBERT for CD^3 with different prompt. The "Ran" is mapped to the random-based approach. The "Jsd" is mapped to the mask-based method. The "Cos" is mapped to the similarity-based method. The suffix "$10 * 5$" denotes beam width $K = 10$. The suffix "$20 * 5$" denotes beam width $K = 20$. The best result is indicated in **bold**.

Model	Biased-Prompt	SEAT-6	SEAT-6b	SEAT-7	SEAT-7b	SEAT-8	SEAT-8b	Avg.esize
BERT $+\text{CD}^3$	Ran_10 ∗ 5	0.4651	-0.0925	-0.6087	0.6223	-0.2971	-0.5076	0.4322
	Jsd_10 ∗ 5	**0.4034**	-0.1781	-0.5684	0.5930	0.0446	-0.4673	0.3758
	Cos_10 ∗ 5	0.5722	**-0.0076**	-0.3527	**0.5201**	-0.3924	-0.4351	0.3800
	Ran_20 ∗ 5	0.5658	-0.0497	-0.5869	0.6935	-0.2426	-0.3805	0.4198
	Jsd_20 ∗ 5	0.6190	-0.0358	-0.5448	0.7885	**-0.0018**	**-0.1771**	0.3612
	Cos_20 ∗ 5	0.6670	0.0298	**-0.0187**	0.5807	-0.2790	-0.2935	**0.3115**
ALBERT $+\text{CD}^3$	Ran_10 ∗ 5	0.1522	-0.2635	-0.1548	-0.2624	0.2278	-0.2095	0.1889
	Jsd_10 ∗ 5	**0.1013**	**-0.1126**	-0.0964	-0.2582	0.1652	0.0253	0.1265
	Cos_10 ∗ 5	0.1408	-0.2292	-0.0462	-0.1091	0.0339	**-0.0058**	0.0942
	Ran_20 ∗ 5	0.1332	-0.2262	-0.1027	-0.2866	0.2022	-0.0970	0.1747
	Jsd_20 ∗ 5	0.1116	-0.1200	-0.0805	-0.2535	0.1313	0.0151	0.1187
	Cos_20 ∗ 5	0.1253	-0.1707	**-0.0195**	**-0.0087**	**0.0233**	0.0636	**0.0685**

5.3.1. Gender debiasing

The gender debiasing experiments are shown in Table 3. In the quantity dimension, we explore the biased prompts with beam width $K = 10$ and $K = 20$ (i.e., the total number of biased prompts is 50 and 100). In the probing method dimension, we explore three ways to probe biases in PLMs' encodings: similarity-based method, mask-based method, and random-based method.

In the similarity-based method as described in Section 4, we use Cosine Similarity to calculate the representation distance between positive sample pairs and then search for the top K biased prompts within each iteration that can cause the maximum differences between the representations distance. The specific generation process is described in Algorithm 1.

The mask-based method adds a placeholder $[MASK]$ to the template as follows:

$$\begin{aligned}(x', \hat{x}') = (&x \oplus (p^1, p^2, \cdots, p^D) \oplus [MASK],\\ &\hat{x} \oplus (p^1, p^2, \cdots, p^D) \oplus [MASK]),\end{aligned} \tag{13}$$

where $[MASK]$ represents a masked token filled with sensitive attribute words of another gender direction. For a given positive sample pair, we concatenate each candidate word in the search space $\mathcal{V}$ and then calculate the prediction result of the masked position. We aim to find the top K candidates within each iteration that maximize the difference between the predicted results among the positive samples as biased prompts. Specifically, we choose the JSD as the target loss function of the prediction result, which is defined as follows:

$$\begin{aligned}&JSD(x', \hat{x}')\\ &= \frac{1}{2}(KLD(q_{x'} || \frac{q_{x'} + q_{\hat{x}'}}{2}) + KLD(q_{\hat{x}'} || \frac{q_{x'} + q_{\hat{x}'}}{2}))\\ &= \frac{1}{2}(\sum_{v \in \mathcal{V}} q_{x'}(v) \log \frac{q_{x'}(v)}{q_{\hat{x}'}(v)} + \sum_{v \in \mathcal{V}} q_{\hat{x}'}(v) \log \frac{q_{\hat{x}'}(v)}{q_{x'}(v)}),\end{aligned} \tag{14}$$

where $q_{x'}$ and $q_{\hat{x}'}$ respectively represent the predicted distribution probability of the $[MASK]$ in sentence x' and $\hat{x}'$.

The random-based method randomly selects a corresponding number of words in the search space $\mathcal{V}$ as biased prompts. The double data augmentation can expand the original training data so that the model can be trained more fully. The reason why we compared the random-based method is to explore whether the same order of magnitude of data expansion plays a role in the debiasing ability.

The SEAT test results are shown in Table 3, the average effect size of the experimental group with biased prompts of $K = 20$ is better than the experimental group with $K = 10$. The ability of CD^3 to mitigate bias increases with the number of biased prompts. It is important to note that although we can expand the magnitude of biased prompts without limit, which can decrease the language modeling capabilities of PLMs. We chose to set the value of $K = 20$ to ensure the debiasing performance while maintaining language modeling capability. We observed that the similarity-based method performs the best overall among the three probing methods. Although excellent in some subtest, the mask-based method performs slightly worse overall. We analyze the reason why the similarity-based method has better experimental results than the mask-based method. The similarity-based method directly measures the distance between sentence representations, omits the step of classifier head prediction and avoids the possible bias within PLMs. In addition, results of the random-based method show that the expanded training data play a small role in debiasing, and its debiasing ability is mainly due to the contrastive self-debiasing stage.

Table 4
Race debiasing results of SEAT on BERT and ALBERT for CD^3 with different prompt. The "Ran" is mapped to the random-based method. The "Cos" is mapped to the similarity-based method. The suffix "$10 * 5$" denotes beam width $K = 10$. The suffix "$20 * 5$" denotes beam width $K = 20$. The best result is indicated in **bold**.

Model	Biased-Prompt	SEAT-3	SEAT-3b	SEAT-4	SEAT-5	SEAT-5b	Avg.esize
BERT + CD^3	Ran_10 ∗ 5	-0.2430	-0.1249	-0.4461	-0.4150	-0.1103	0.2679
	Cos_10 ∗ 5	-0.2221	**0.0367**	-0.3821	-0.3745	**0.0621**	0.2155
	Ran_20 ∗ 5	-0.2300	-0.1221	-0.4144	-0.4310	-0.1203	0.2636
	Cos_20 ∗ 5	**-0.0657**	-0.1280	**-0.3783**	**-0.3549**	-0.1291	**0.2112**
ALBERT + CD^3	Ran_10 ∗ 5	0.6001	0.0729	0.0910	0.1064	0.1203	0.1981
	Cos_10 ∗ 5	**0.4453**	-0.0409	-0.0650	-0.0948	-0.0187	0.1330
	Ran_20 ∗ 5	0.5787	0.0676	0.0810	0.1032	0.1162	0.1893
	Cos_20 ∗ 5	0.5005	**-0.0136**	**0.0128**	**-0.0256**	**0.0023**	**0.1110**

5.3.2. *Race debiasing*

For the race debiasing experiments, we report the SEAT test statistics of CD^3 with different prompts in Table 4. Similar to gender debiasing, in the quantity dimension, K is also selected as 10 and 20. In the probing method dimension, we only explore the random-based method and the similarity-based method. Their implementation details are the same as gender debiasing.

It can be observed from the Table 4 that the race debiasing results in BERT and ALBERT show the same trend as in the gender experiments. The debiasing performance of CD^3 increases with the number of biased prompts, and the random-based method does not show a good effect.

In summary, CD^3 shows the same trend in the results of gender and race debiasing for BERT and ALBERT. The quantity of biased prompt significantly impacts the debiasing performance of CD^3, and the higher the order of magnitude, the more attention the model pays to social biases. Different probing methods also have different effects on the debiasing performance of CD^3. Ablation studies have answered the second question (**Q2**) proposed at the beginning of this section.

5.4. *Performance on downstream NLU tasks*

To verify that our proposed debiasing model retains the original language modeling capabilities of PLMs, we also measure the model's performance on the General Language Understanding Evaluation (GLUE) [72] downstream tasks. For all fine-tuning experiments, we set the batch size to 32 and the learning rate to 2×10^{-5}. The experimental results on BERT and ALBERT are shown in Table 5.

Overall, most downstream tasks are unaffected by debiasing. There are a few special cases where Dropout shows a wide gap from the original result in COLA task of two PLMs and RET task of ALBERT. We analyze that Dropout changing the internal structure of PLMs is the direct cause of impairing its language modeling ability. In addition, we find that the results on WNLI show instability, which may be due to the small size of the task besides the effect of debiasing. The other baseline models perform at the same level as BERT and ALBERT and even score slightly better than the original language model on some downstream tasks. It should be noted that FairFil, Auto-Debias, INLP, and CD^3 all surpass ALBERT in the scoring average. This suggests that the debiasing technique does not compromise the language modeling capabilities of PLMs, or that the fine-tuning process relearns parameters beneficial to downstream tasks.

In summary, CD^3 shows performance comparable to the original language model for all downstream tasks. CD^3 reduces the social biases of PLMs while maintaining the language modeling capability in the downstream tasks. We have answered the third question (**Q3**) proposed at the beginning of this section.

5.5. *Analysis of biased prompts*

We provide some examples of biased prompts in Table 6, which are automatically searched by the double data augmentation on BERT. We present the results for each of the five iterations targeting gender bias and race bias, respectively. From the results, some of the biased prompts appear to be bias-related. For example, "heavyweight," "he," "pitcher," and "men" may have stereotypical associations with gender group. Some of the more implicit associations, such as "admitted", "refused", "pointed", and "coach" seem to carry a certain emotional predisposition to race group. The other tokens are uninterpretable, which is not surprising since the search for discrete prompts is not based on syntax and interpretability. And due to the large size and diversity of the pre-trained corpus, as well as the black-box nature of the model, it is difficult to determine the distribution of social biases that the model obtains from the pre-trained corpus. This further confirms the necessity of our proposed double data augmentation, as these unexplained tokens are difficult to detect and define manually. CD^3 not only saves labor costs but also has a more significant debiasing effect.

5.6. *Time complexity analysis of double data augmentation*

In the double data augmentation stage, when cos similarity is used to measure the distance between positive sample pairs, the time complexity can be calculated as follows:

Table 5
Experimental results of GLUE tasks on BERT and ALBERT. 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.

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	55.97	84.84	87.22	90.84	87.85	63.29	92.32	88.43	53.66	78.27
+Dropout	50.87	84.78	88.22	91.49	88.02	62.29	92.09	87.87	51.66	77.48
+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	50.70	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
+CD^3 (ours)	55.96	84.91	88.28	91.65	87.81	65.34	92.43	88.44	54.39	78.80
ALBERT	55.61	85.38	90.85	91.21	88.99	72.56	92.32	89.39	39.44	78.42
+CDA	53.38	84.90	88.86	91.20	87.35	62.77	91.28	87.34	50.34	77.49
+Dropout	46.83	85.13	89.83	90.99	86.90	55.71	91.86	88.68	52.34	76.47
+Sent-Debias	56.37	84.96	91.65	91.92	87.56	67.87	92.31	90.46	33.80	77.43
+FairFil	55.85	86.09	91.20	91.84	87.47	73.29	93.12	90.22	46.48	79.51
+Auto-Debias	56.69	79.25	91.61	91.94	87.41	73.29	93.35	90.48	54.93	79.88
+INLP	58.18	85.63	90.81	91.69	87.50	71.12	92.20	90.68	52.11	79.99
+CD^3 (ours)	55.42	85.24	89.70	91.63	87.28	72.56	92.66	89.88	46.48	78.98

Table 6
Examples of biased prompts automatically searched by the double data augmentation on BERT.

Bias	Iterations	Biased Prompts
Gender	1	heavyweight, he, franchise, exactly, pitcher, typical
	2	heavyweight practices, initially men, initially harvard
	3	he initially strange, initially men experiment
	4	he initially experienced traveling, he initially twice trains
	5	he initially experienced sailing internationally
Race	1	admitted, refused, stopped, informed, pointed, challenges
	2	challenges either, asked approximately, either coach
	3	informed either whether, challenges either allows
	4	started either approximately whether, either coach plays or
	5	challenges either allows mechanism writes

$$(|\mathcal{V}| \times |\mathcal{N}|)^2 + ((\epsilon - 1) \times K \times |V| \times |\mathcal{N}|)^2, \tag{15}$$

where the first term represents the time complexity of the first iterative search in beam search, the second term represents the time complexity of the remaining iterative search, and the quadratic power represents the calculation of cos similarity. $|\mathcal{V}|$ is thesearch space size, $|\mathcal{N}|$ is the training dataset size in the data augmentation stage, K is the beam width, and ϵ is the number of iterations. When the training dataset is large enough, the influence of search space size, beam width and number of iterations can be ignored, and the time complexity can be approximately $\mathcal{O}(|\mathcal{N}|^2)$.

In the experimental implementation, due to the small training dataset we choose, the execution time is affected by the search vocabulary, beam width, and the number of iterations. When $|\mathcal{V}|$ is 5000, K is 20, and ϵ is 5, the execution time of the double data augmentation on four RTX 3090 GPUs is approximately 6 hours. When K is 10, the execution time is roughly halved. The double data augmentation stage is performed once and does not need to be deployed to a later debiasing stage. Therefore, we consider this time cost to be acceptable.

6. Conclusions

This paper mitigates the social biases specific to demographic groups of pre-trained language models. We propose a contrastive self-debiasing model with double data augmentation (CD^3) consisting of two stages. In the first stage, CD^3 applies double data augmentation, augmenting the original training data and amplifying the social biases between different demographic groups. Double data augmentation utilizes automatically searched biased prompts to reduce the limitation of manual prior knowledge. In the second stage, CD^3 uses the augmented training data for contrastive learning to train a plug-and-play self-debiasing adapter. Contrastive self-debiasing mitigates social biases in encoding while preserving language modeling capabilities without changing PLMs. We have carried out a large number of experiments on BERT, ALBERT, and RoBERTa, the results verify that our model is better than the state-of-the-art model in gender debiasing and race debiasing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgements

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. 62372211), the International Science and Technology Cooperation Program of Jilin Province (No. 20230402076GH, No. 20240402067GH), the Science and Technology Development Program of Jilin Province (No. 20220201153GX).

References

[1] J. Su, J. Tang, H. Jiang, Z. Lu, Y. Ge, L. Song, D. Xiong, L. Sun, J. Luo, Enhanced aspect-based sentiment analysis models with progressive self-supervised attention learning, Artif. Intell. 296 (2021) 103477.

[2] K.V. Deshpande, S. Pan, J.R. Foulds, Mitigating demographic bias in ai-based resume filtering, in: Proc. 28th UMAP Adjun. - Adjun. Publ. ACM Conf. User Model., Adapt. Pers., ACM, 2020, pp. 268–275.

[3] Z. Obermeyer, B. Powers, C. Vogeli, S. Mullainathan, Dissecting racial bias in an algorithm used to manage the health of populations, Science 366 (6464) (2019) 447–453.

[4] N. Garg, L. Schiebinger, D. Jurafsky, J. Zou, Word embeddings quantify 100 years of gender and ethnic stereotypes, Proc. Natl. Acad. Sci. 115 (16) (2018) E3635–E3644.

[5] R. Liu, C. Jia, J. Wei, G. Xu, S. Vosoughi, Quantifying and alleviating political bias in language models, Artif. Intell. 304 (2022) 103654.

[6] F. Wu, M. Du, C. Fan, R. Tang, Y. Yang, A. Mostafavi, X. Hu, Understanding social biases behind location names in contextual word embedding models, IEEE Trans. Comput. Soc. Syst. 9 (2) (2021) 458–468.

[7] T. Bolukbasi, K. Chang, J.Y. Zou, V. Saligrama, A.T. Kalai, Man is to computer programmer as woman is to homemaker? Debiasing word embeddings, in: Proc. 30th Adv. Neural Inf. Proces. Syst., 2016, pp. 4349–4357.

[8] A. Caliskan, J.J. Bryson, A. Narayanan, Semantics derived automatically from language corpora contain human-like biases, Science 356 (6334) (2017) 183–186.

[9] E. Sheng, K. Chang, P. Natarajan, N. Peng, Societal biases in language generation: progress and challenges, in: Proc. 59th Annu. Meet. Assoc. Comput. Linguist. Int. Jt. Conf. Nat. Lang. Process., 2021, pp. 4275–4293.

[10] K. Webster, X. Wang, I. Tenney, A. Beutel, E. Pitler, E. Pavlick, J. Chen, S. Petrov, Measuring and reducing gendered correlations in pre-trained models, CoRR, arXiv:2010.06032 [abs].

[11] M. Zhang, U. Niranjan, Y. He, Adversarial data augmentation for task-specific knowledge distillation of pre-trained transformers, in: Proc. 36th AAAI Conf. Artif. Intell., 2022, pp. 11685–11693.

[12] Y. Wang, C. Xu, Q. Sun, H. Hu, C. Tao, X. Geng, D. Jiang, Promda: prompt-based data augmentation for low-resource NLU tasks, in: Proc. 60th Annu. Meeting Assoc. Comput. Linguistics, 2022, pp. 4242–4255.

[13] R. Zmigrod, S.J. Mielke, H.M. Wallach, R. Cotterell, Counterfactual data augmentation for mitigating gender stereotypes in languages with rich morphology, in: Proc. 57th Annu. Meeting Assoc. Comput. Linguistics, 2019, pp. 1651–1661.

[14] H. Chen, R. Xia, J. Yu, Reinforced counterfactual data augmentation for dual sentiment classification, in: Proc. Conf. Empir. Methods Natural Lang. Process., 2021, pp. 269–278.

[15] A. Garimella, A. Amarnath, K. Kumar, A.P. Yalla, A. Natarajan, N. Chhaya, B.V. Srinivasan, He is very intelligent, she is very beautiful? On mitigating social biases in language modelling and generation, in: Proc. 59th Annu. Meeting Assoc. Comput. Linguistics, 2021, pp. 4534–4545.

[16] J. Devlin, M. Chang, K. Lee, K. Toutanova, BERT: pre-training of deep bidirectional transformers for language understanding, in: Proc. Conf. N. Am. Chapter Assoc. Comput. Linguist.: Hum. Lang. Technol., 2019, pp. 4171–4186.

[17] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, R. Soricut, ALBERT: a lite BERT for self-supervised learning of language representations, in: Proc. 8th Int. Conf. Learn. Represent., 2020.

[18] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, Roberta: a robustly optimized BERT pretraining approach, CoRR, arXiv:1907.11692 [abs].

[19] M. Kaneko, D. Bollegala, Unmasking the mask - evaluating social biases in masked language models, in: Proc. 36th AAAI Conf. Artif. Intell., 2022, pp. 11954–11962.

[20] Y.C. Tan, L.E. Celis, Assessing social and intersectional biases in contextualized word representations, in: Proc. 33th Adv. Neural Inf. Proces. Syst., 2019, pp. 13209–13220.

[21] L. Dixon, J. Li, J. Sorensen, N. Thain, L. Vasserman, Measuring and mitigating unintended bias in text classification, in: Proc. 1st AAAI/ACM Conf. AI, Ethics, Soc., 2018, pp. 67–73.

[22] S. Jentzsch, P. Schramowski, C. Rothkopf, K. Kersting, Semantics derived automatically from language corpora contain human-like moral choices, in: Proc. 2nd AAAI/ACM Conf. AI, Ethics, Soc., 2019, pp. 37–44.

[23] Y. Li, M. Du, R. Song, X. Wang, Y. Wang, A survey on fairness in large language models, CoRR, arXiv:2308.10149 [abs].

[24] Y.T. Cao, Y. Pruksachatkun, K. Chang, R. Gupta, V. Kumar, J. Dhamala, A. Galstyan, On the intrinsic and extrinsic fairness evaluation metrics for contextualized language representations, in: Proc. 60th Annu. Meeting Assoc. Comput. Linguistics, 2022, pp. 561–570.

[25] C. May, A. Wang, S. Bordia, S.R. Bowman, R. Rudinger, On measuring social biases in sentence encoders, in: Proc. Conf. N. Am. Chapter Assoc. Comput. Linguist.: Hum. Lang. Technol., 2019, pp. 622–628.

[26] N. Nangia, C. Vania, R. Bhalerao, S.R. Bowman, Crows-pairs: a challenge dataset for measuring social biases in masked language models, in: Proc. Conf. Empir. Methods Natural Lang. Process., 2020, pp. 1953–1967.

[27] D. Borkan, L. Dixon, J. Sorensen, N. Thain, L. Vasserman, Nuanced metrics for measuring unintended bias with real data for text classification, in: Web Conf. - Companion World Wide Web Conf., WWW, ACM, 2019, pp. 491–500.

[28] T. Sun, A. Gaut, S. Tang, Y. Huang, M. ElSherief, J. Zhao, D. Mirza, E.M. Belding, K. Chang, W.Y. Wang, Mitigating gender bias in natural language processing: literature review, in: Proc. 57th Annu. Meeting Assoc. Comput. Linguistics, 2019, pp. 1630–1640.

[29] M. Kim, J. Kim, K.M. Johnson, Race, gender, and age biases in biomedical masked language models, in: Proc. 61st Findings of Annu. Meeting Assoc. Comput. Linguistics, 2023, pp. 11806–11815.

[30] K. Khadilkar, A.R. KhudaBukhsh, An unfair affinity toward fairness: characterizing 70 years of social biases in bhollywood (student abstract), in: Proc. 35th AAAI Conf. Artif. Intell., 2021, pp. 15813–15814.

[31] P. Sen, D. Ganguly, Towards socially responsible AI: cognitive bias-aware multi-objective learning, in: Proc. 34th AAAI Conf. Artif. Intell., 2020, pp. 2685–2692.

[32] C. Amrhein, F. Schottmann, R. Sennrich, S. Läubli, Exploiting biased models to de-bias text: a gender-fair rewriting model, in: Proc. 61st Annu. Meeting Assoc. Comput. Linguistics, 2023, pp. 4486–4506.

[33] M.L. Olson, R. Khanna, L. Neal, F. Li, W. Wong, Counterfactual state explanations for reinforcement learning agents via generative deep learning, Artif. Intell. 295 (2021) 103455.

[34] J. Zhao, T. Wang, M. Yatskar, V. Ordonez, K. Chang, Gender bias in coreference resolution: evaluation and debiasing methods, in: Proc. Conf. N. Am. Chapter Assoc. Comput. Linguistics: Hum. Lang. Technol., 2018, pp. 15–20.

[35] J. He, M. Xia, C. Fellbaum, D. Chen, MABEL: attenuating gender bias using textual entailment data, in: Proc. Conf. Empir. Methods Natural Lang. Process., 2022, pp. 9681–9702.

[36] Q. Liu, M.J. Kusner, P. Blunsom, Counterfactual data augmentation for neural machine translation, in: Proc. Conf. N. Am. Chapter Assoc. Comput. Linguist.: Hum. Lang. Technol., 2021, pp. 187–197.

[37] Y. Li, M. Du, X. Wang, Y. Wang, Prompt tuning pushes farther, contrastive learning pulls closer: a two-stage approach to mitigate social biases, in: Proc. 61st Annu. Meeting Assoc. Comput. Linguistics, 2023, pp. 14254–14267.

[38] A. Omrani, A.S. Ziabari, C. Yu, P. Golazizian, B. Kennedy, M. Atari, H. Ji, M. Dehghani, Social-group-agnostic bias mitigation via the stereotype content model, in: Proc. 61st Annu. Meeting Assoc. Comput. Linguistics, 2023, pp. 4123–4139.

[39] M. Kaneko, D. Bollegala, Debiasing pre-trained contextualised embeddings, in: Proc. 16th Conf. Eur. Chapter Assoc. Comput. Linguist., 2021, pp. 1256–1266.

[40] Y. Guo, Y. Yang, A. Abbasi, Auto-debias: debiasing masked language models with automated biased prompts, in: Proc. 60th Annu. Meeting Assoc. Comput. Linguistics, 2022, pp. 1012–1023.

[41] T. Manzini, Y.C. Lim, A.W. Black, Y. Tsvetkov, Black is to criminal as caucasian is to police: detecting and removing multiclass bias in word embeddings, in: Proc. Conf. N. Am. Chapter Assoc. Comput. Linguist.: Hum. Lang. Technol., 2019, pp. 615–621.

[42] R. Song, F. Giunchiglia, Y. Li, L. Shi, H. Xu, Measuring and mitigating language model biases in abusive language detection, Inf. Process. Manag. 60 (3) (2023) 103277.

[43] S. Ravfogel, Y. Elazar, H. Gonen, M. Twiton, Y. Goldberg, Null it out: guarding protected attributes by iterative nullspace projection, in: Proc. 58th Annu. Meeting Assoc. Comput. Linguistics, 2020, pp. 7237–7256.

[44] P.P. Liang, I.M. Li, E. Zheng, Y.C. Lim, R. Salakhutdinov, L. Morency, Towards debiasing sentence representations, in: Proc. 58th Annu. Meeting Assoc. Comput. Linguistics, 2020, pp. 5502–5515.

[45] P. Cheng, W. Hao, S. Yuan, S. Si, L. Carin, Fairfil: contrastive neural debiasing method for pretrained text encoders, in: Proc. 9th Int. Conf. Learn. Represent, 2021.

[46] A. Lauscher, T. Lüken, G. Glavas, Sustainable modular debiasing of language models, in: Proc. Findings of Conf. Empir. Methods Natural Lang. Process., 2021, pp. 4782–4797.

[47] Y. Gu, X. Han, Z. Liu, M. Huang, PPT: pre-trained prompt tuning for few-shot learning, in: Proc. 60th Annu. Meeting Assoc. Comput. Linguistics, 2022, pp. 8410–8423.

[48] T. Shin, Y. Razeghi, R.L.L. IV, E. Wallace, S. Singh, Autoprompt: eliciting knowledge from language models with automatically generated prompts, in: Proc. Conf. Empir. Methods Natural Lang. Process., 2020, pp. 4222–4235.

[49] A. Webson, E. Pavlick, Do prompt-based models really understand the meaning of their prompts?, in: Proc. Conf. N. Am. Chapter Assoc. Comput. Linguist.: Hum. Lang. Technol., 2022, pp. 2300–2344.

[50] J. Liu, A. Liu, X. Lu, S. Welleck, P. West, R.L. Bras, Y. Choi, H. Hajishirzi, Generated knowledge prompting for commonsense reasoning, in: Proc. 60th Annu. Meeting Assoc. Comput. Linguistics, 2022, pp. 3154–3169.

[51] P. Liu, W. Yuan, J. Fu, Z. Jiang, H. Hayashi, G. Neubig, Pre-train, prompt, and predict: a systematic survey of prompting methods in natural language processing, CoRR, arXiv:2107.13586 [abs].

[52] P.A.M. Casares, B.S. Loe, J. Burden, S.Ó. hÉigeartaigh, J. Hernández-Orallo, How general-purpose is a language model? Usefulness and safety with human prompters in the wild, in: Proc. 36th AAAI Conf. Artif. Intell., 2022, pp. 5295–5303.

[53] T. Chen, S. Kornblith, M. Norouzi, G.E. Hinton, A simple framework for contrastive learning of visual representations, in: Proc. 37th Int. Conf. Mach. Learn, 2020, pp. 1597–1607.

[54] E.A. Haroutunian, Information theory and statistics, in: International Encyclopedia of Statistical Science, 2011, pp. 666–667.

[55] R. Han, W. Wang, Y. Long, J. Peng, Deep representation debiasing via mutual information minimization and maximization (student abstract), in: Proc. 36th AAAI Conf. Artif. Intell., 2022, pp. 12965–12966.

[56] Z. Zeng, D. Xiong, Unsupervised and few-shot parsing from pretrained language models, Artif. Intell. 305 (2022) 103665.

[57] M. Gutmann, A. Hyvärinen, Noise-contrastive estimation: a new estimation principle for unnormalized statistical models, in: Proc. 30th Int. Conf. Artif. Intell. Stat., 2010, pp. 297–304.

[58] M. Nadeem, A. Bethke, S. Reddy, Stereoset: measuring stereotypical bias in pretrained language models, in: Proc. 59th Annu. Meet. Assoc. Comput. Linguist. Int. Jt. Conf. Nat. Lang. Process., 2021, pp. 5356–5371.

[59] S. Park, H.S. Shim, M. Chatterjee, K. Sagae, L. Morency, Computational analysis of persuasiveness in social multimedia: a novel dataset and multimodal prediction approach, in: Proc. 16th ACM Int. Conf. Proc. Ser., 2014, pp. 50–57.

[60] S. Poria, D. Hazarika, N. Majumder, G. Naik, E. Cambria, R. Mihalcea, MELD: a multimodal multi-party dataset for emotion recognition in conversations, in: Proc. 57th Annu. Meeting Assoc. Comput. Linguistics, 2019, pp. 527–536.

[61] S. Merity, C. Xiong, J. Bradbury, R. Socher, Pointer sentinel mixture models, in: Proc. 5th Int. Conf. Learn. Represent., 2017.

[62] M. Völske, M. Potthast, S. Syed, B. Stein, Tl, dr: mining reddit to learn automatic summarization, in: Proc. Workshop New Front. Summ., 2017, pp. 59–63.

[63] R. Socher, A. Perelygin, J. Wu, J. Chuang, C.D. Manning, A.Y. Ng, C. Potts, Recursive deep models for semantic compositionality over a sentiment treebank, in: Proc. Conf. Empir. Methods Natural Lang. Process., 2013, pp. 1631–1642.

[64] S. Desai, G. Durrett, Calibration of pre-trained transformers, in: Proc. Conf. Empir. Methods Natural Lang. Process., 2020, pp. 295–302.

[65] Z. Jiang, F.F. Xu, J. Araki, G. Neubig, How can we know what language models know, Trans. Assoc. Comput. Linguist. 8 (2020) 423–438.

[66] J. Salazar, D. Liang, T.Q. Nguyen, K. Kirchhoff, Masked language model scoring, in: Proc. 58th Annu. Meeting Assoc. Comput. Linguistics, 2020, pp. 2699–2712.

[67] N. Meade, E. Poole-Dayan, S. Reddy, An empirical survey of the effectiveness of debiasing techniques for pre-trained language models, in: Proc. 60th Annu. Meeting Assoc. Comput. Linguistics, 2022, pp. 1878–1898.

[68] S.J. Mason, N.E. Graham, Areas beneath the relative operating characteristics (roc) and relative operating levels (rol) curves: statistical significance and interpretation, Q. J. R. Meteorol. Soc. 128 (584) (2002) 2145–2166.

[69] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. von Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T.L. Scao, S. Gugger, M. Drame, Q. Lhoest, A.M. Rush, Transformers: state-of-the-art natural language processing, in: Proc. Conf. Empir. Methods Natural Lang. Process., 2020, pp. 38–45.

[70] V. Aribandi, Y. Tay, D. Metzler, How reliable are model diagnostics?, in: Find. Assoc. Comput. Linguist.: ACL-IJCNLP, 2021, pp. 1778–1785.

[71] S.L. Blodgett, G. Lopez, A. Olteanu, R. Sim, H.M. Wallach, Stereotyping norwegian salmon: an inventory of pitfalls in fairness benchmark datasets, in: Proc. 59th Annu. Meet. Assoc. Comput. Linguist. Int. Jt. Conf. Nat. Lang. Process., 2021, pp. 1004–1015.

[72] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, S.R. Bowman, GLUE: a multi-task benchmark and analysis platform for natural language understanding, in: Proc. 7th Int. Conf. Learn. Represent., 2019.