Paper ID: 9526

ADEPT: A DEbiasing PrompT Framework

Ke Yang[1], Charles Yu[2], Yi R. Fung[2], Manling Li[2], Heng Ji[2]
Tsinghua University[1], UIUC[2]

Motivation

Biases exist and occur throughout the Natural Language Processing (NLP) lifecycle[1]:

Many real-world tasks have been automated by the application of NLP systems.

- Legal information extraction[2];
- Resume filtering[3];
- General language assistants[4], ...
- Pre-trained language models (PLMs) can be debiased to enable applications that may be inadvertently influenced by the PLM's implicit stereotypes.

Debiasing in the finetuning setting:

A finetuning debiasing method typically puts forward specific loss terms to guide a PLM to remove biases in itself[5].

[1] Blodgett S L, Barocas S, Daumé III H, et al. Language (technology) is power: A critical survey of" bias" in nlp[J].
[2] Rabelo J, Goebel R, Kim M Y, et alH. Overview and Discussion of the Competition on Legal Information Extraction/Entailment (COLIEE) 2021[J].
[3] Abdollahnejad E, Kalman M, Far B H. A Deep Learning BERT-Based Approach to Person-Job Fit in Talent Recruitment[C]2021.
[4] Askell A, Bai Y, Chen A, et al. A general language assistant as a laboratory for alignment[J].
[5] Kaneko M, Bollegala D. Debiasing pre-trained contextualised embeddings[J].

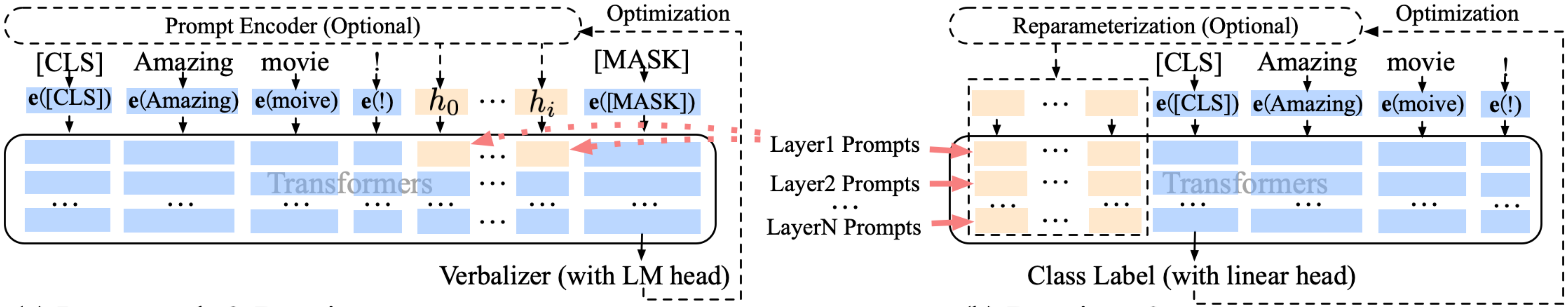
Motivation

A broad experiment of Google BIG-bench[1] shows:

Bias can potentially be steered through appropriately chosen **prompting**.

- In the work of Askell et al. (2021), the authors use **a hand-designed prompt (with more than 4600 solid words)** as a stronger baseline for helpfulness, harmlessness, and honesty.
- With unfixed mathematical representation at the token level, continuous prompts usually surpass discrete ones in providing the models with task-specific supplementary information.

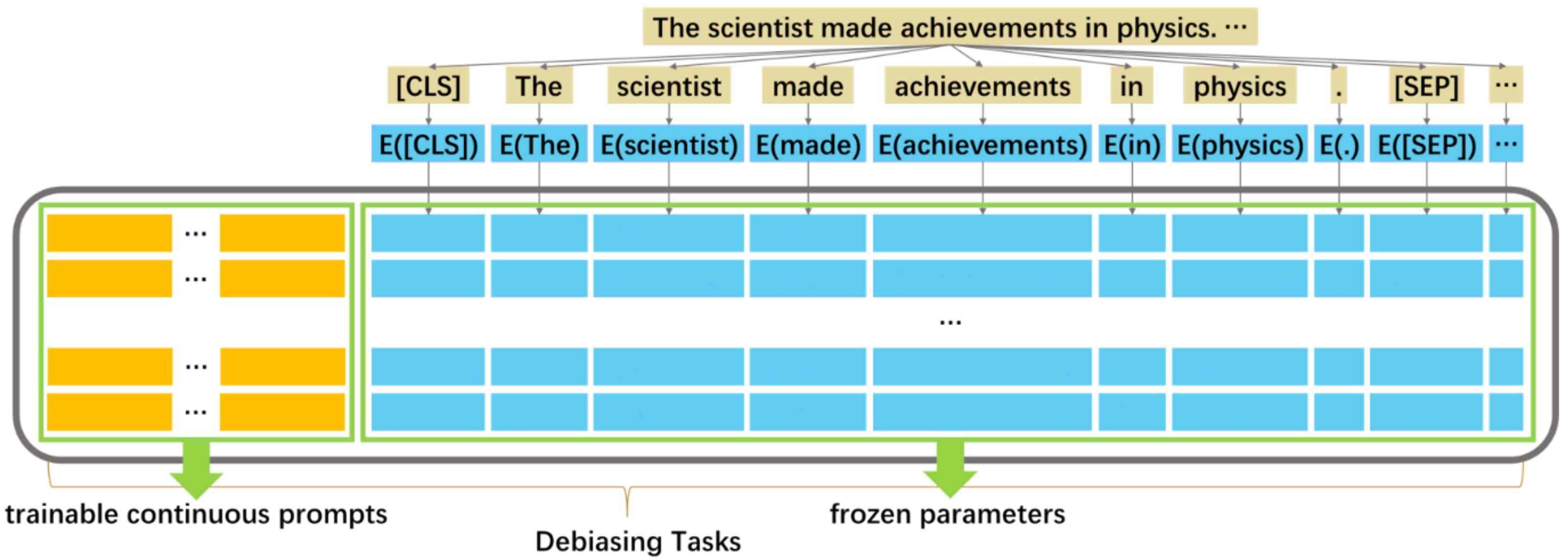
Prompt tuning[2] these days:



(a) Lester et al. & P-tuning (Frozen, 10-billion-scale, simple tasks) (b) P-tuning v2 (Frozen, most scales, most tasks)

[1] Chambers, D., 2018. Tourism research: Beyond the imitation game. Tourism management perspectives, 25, pp.193-195.
[2] Liu, X., Ji, K., Fu, Y., Du, Z., Yang, Z. and Tang, J., 2021. P-tuning v2: Prompt tuning can be comparable to fine-tuning universally across scales and tasks. arXiv preprint arXiv:2110.07602.

Why do we use prompt tuning in debiasing space?



- **It saves computing and storage resources;**
- **It only trains prompt, and the PLM's original parameters are not touched during the training process, so the base model will maintain its robustness;**
- **Continuous prompts in prompt tuning can be optimized with standard techniques like gradient descent.**

Motivation

All pre-trained language model (PLM) debiasing methods must overcome a major hurdle of "imbalance."

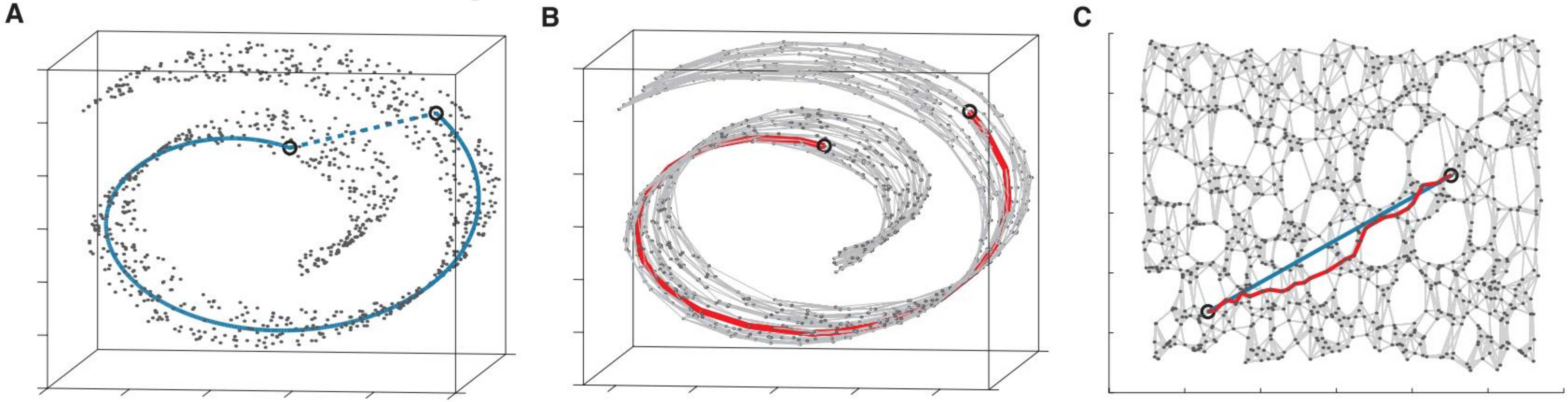
Here, "imbalance" refers to having a hard time keeping the balance between bias mitigation and expressiveness maintenance.

- Existing debiasing methods tend to be "destructive":
  - [1] reduces a word/sentence embedding's projection on a linear bias subspace;
  - [2] completely removes the semantic meanings of attribute words (e.g., man, male; and woman, female) from neutral words (e.g., engineer, scientist; and teacher, librarian).
- Improper debiasing methods may counteract the benefits of pre-training altogether:
  - Although an extreme example, a randomly initialized model is expected to be completely unbiased.

[1] Liang P P, Li I M, Zheng E, et al. Towards debiasing sentence representations[J]. arXiv preprint arXiv:2007.08100, 2020.
[2] Kaneko M, Bollegala D. Debiasing pre-trained contextualised embeddings[J]. arXiv preprint arXiv:2101.09523, 2021.

Manifold Learning

"**Manifold learning** is a popular and quickly-growing subfield of machine learning based on the assumption that one's observed data lie on **a low-dimensional manifold embedded in a higher-dimensional space**." quoted from [1].

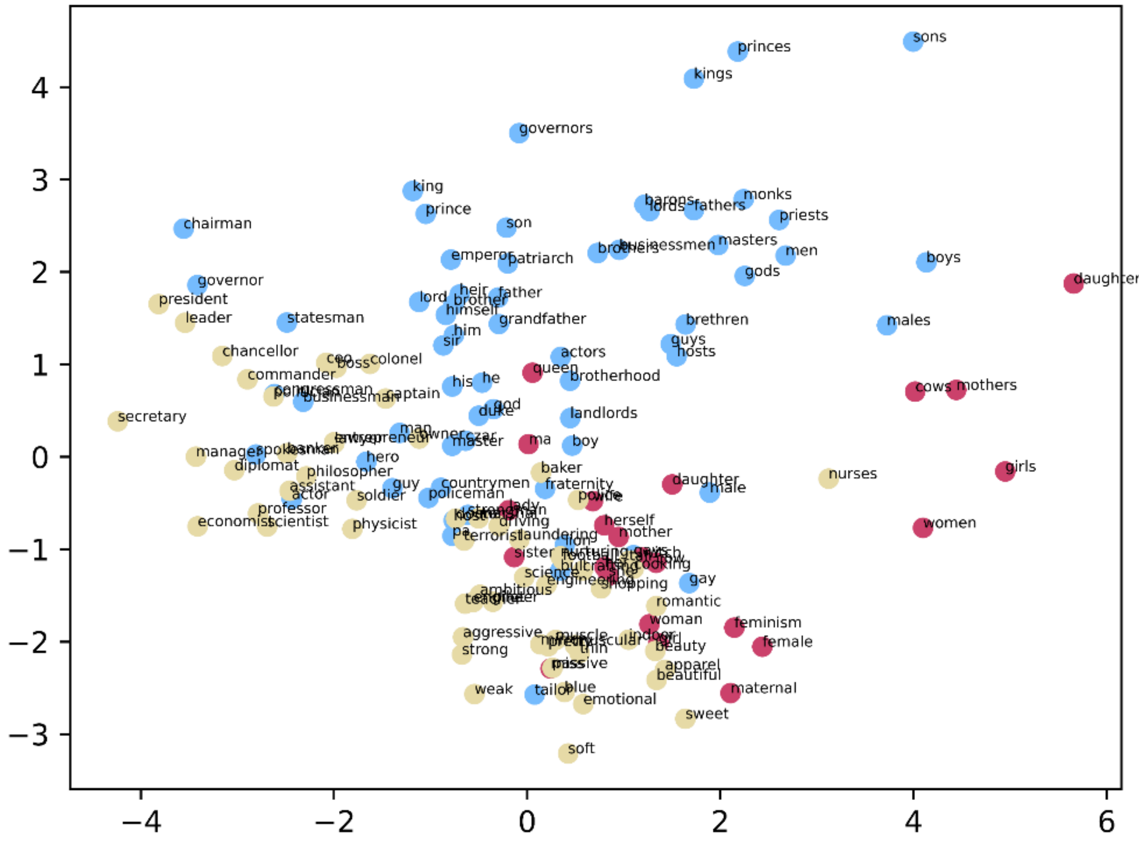


[1] Izenman, A.J., 2012. Introduction to manifold learning. Wiley Interdisciplinary Reviews: Computational Statistics, 4(5), pp.439-446.

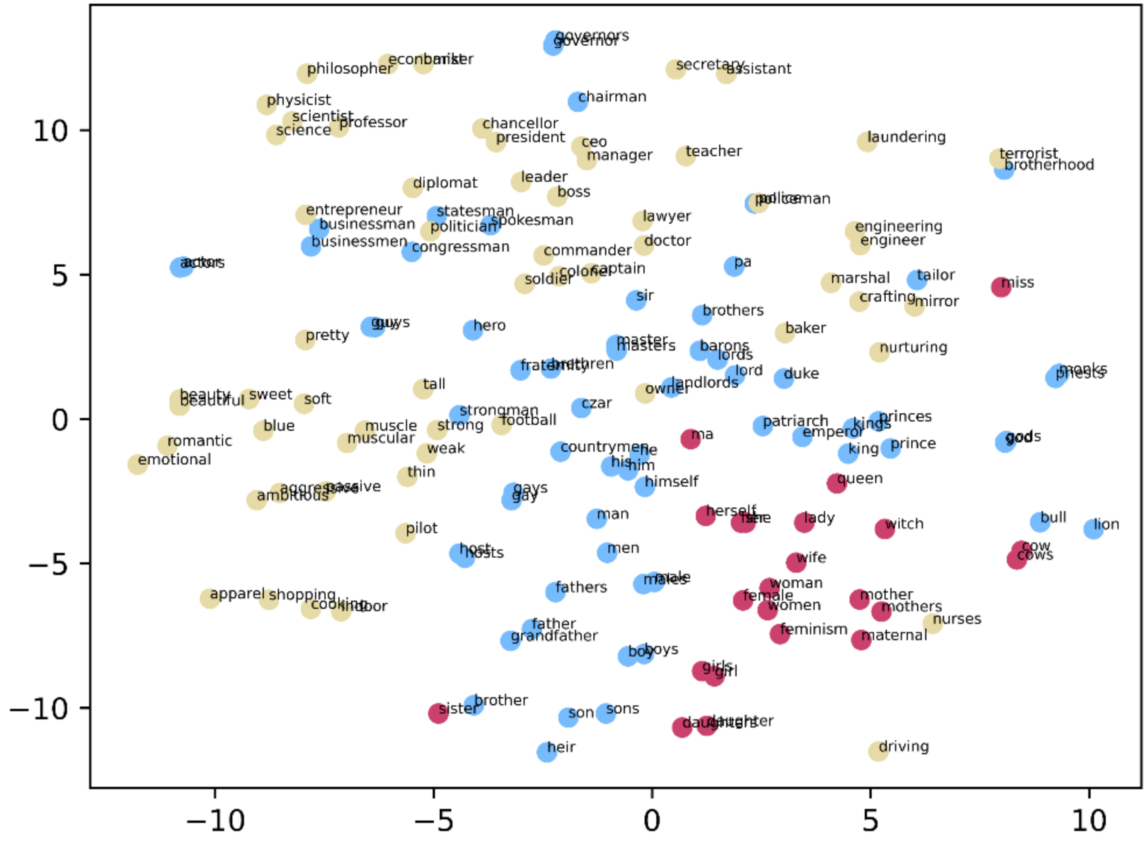
[2] Tenenbaum, J.B., Silva, V.D. and Langford, J.C., 2000. A global geometric framework for nonlinear dimensionality reduction. science, 290(5500), pp.2319-2323.

Linear Assumption VS Manifold Assumption

PCA



t-SNE



Sky blue for masculine, dark pink for feminine, and beige for neuter words. These word sets are defined in the paper[1].

Compared with the one depicted by PCA under the globally linear assumption, the one using **t-SNE**, following the manifold learning idea, shows a **clearer correlation between pairwise words**.

[1] Kaneko, M. and Bollegala, D., 2021. Debiasing pre-trained contextualised embeddings. arXiv preprint arXiv:2101.09523.

Task Formulation

- Our goal is: given a PLM $\boldsymbol{M_\Theta}$ with parameter Θ, find the parameters Φ_{prompt} determining a set of continuous prompts, so that the prompt-tuned model $M_{\Theta\cup\Phi_{prompt}}$ (we will use $\mathbf{M}'_{\Theta}$ for short) **has the debiasing effects while maintaining the expressiveness of M_Θ**.
- We optimize Φ_{prompt} by using the objective function:

$$\boldsymbol{L = L_{bias} + \lambda * L_{representation}}$$

where $\boldsymbol{L_{bias}}$ seeks to minimize biases in M'_{Θ} whereas $\boldsymbol{L_{representation}}$ caters to the debiased model's expressiveness.

Algorithm

Algorithm 1: **ADEPT**: a debiasing algorithm for contextualized word embeddings.

Input: a Pretrained Language Model (PLM)
Output: Φ_{prompt} for debiasing the PLM
ADEPT:

Prepare a PLM M_Θ with parameters Θ.
Suppose a bias has d attributes. Define a neutral word tuple $W^{neutral}$ and attribute word tuples $W^{a(i)} = (w_1^{a(i)}, ..., w_g^{a(i)})$, each with g one-to-one words.
Collect sentences $S^{neutral}$ and $\{S^{a(i)}\}_{i=1}^{d}$.
Initialize parameters Φ_{prompt}.
for epoch in 1, ..., $epoch_{max}$ **do**
  Calculate prototypes of the neutral words:
  $E^{neutral} = M'_\Theta(S^{neutral})$,
  where $M'_\Theta = M_{\Theta \cup \Phi_{prompt}}$.
  Calculate prototypes of attributes:
  $E^{a(i)} = M'_\Theta(S^{a(i)}), e^{a(i)} = aver(E^{a(i)})$.
  Calculate distances between attribute words and neutral words: $P^{a(i)} = Distance(E^{neutral}|e^{a(i)})$.
  Calculate loss of bias:
  $L_{bias} = \sum_{i,j \in \{1,...,d\}} \{JS(P^{a(i)}||P^{a(j)})\}$.
  Calculate loss of representation:
  $L_{representation} = KL(M_\Theta(S)||M'_\Theta(S))$,
  where $S = S^{neutral} \cup \{S^{a(i)}\}_{i=1}^{d}$.
  Calculate the total loss:
  $L = L_{bias} + \lambda L_{representation}$.
  Compute gradient.
  Update Φ_{prompt}.
end for
return best Φ_{prompt}

Define Word Tuples and Collect Sentences

Here, we obtain $W^{neutral}, W^{a(i)}, S^{neutral}, \{S^{a(i)}\}_{i=1}^{d}$. Toy examples in the *binary gender setting*[1]:

- $W^{neutral}$ = ("*engineer*", "*scientist*", "*teacher*", "*librarian*")
- W^{male} = ("*uncle*", "*father*", "*brother*")
- W^{female} = ("*aunt*", "*mother*", "*sister*")
- $S^{neutral}$ = {"*Engineers are professionals.*", "*Teachers help students acquire knowledge.*", ...}
- S^{male} and S^{female} denote likewise.

Calculate Prototypes of Neutral Words/Attributes

Here, we calculate $E^{neutral}$ and $e^{a(i)}$:

- $E^{neutral} = M'_\Theta(S^{neutral}) = [e_1^{neutral}, e_2^{neutral}, ...]$
- $E^{a(i)} = M'_\Theta(S^{a(i)}), e^{a(i)} = aver(E^{a(i)})$

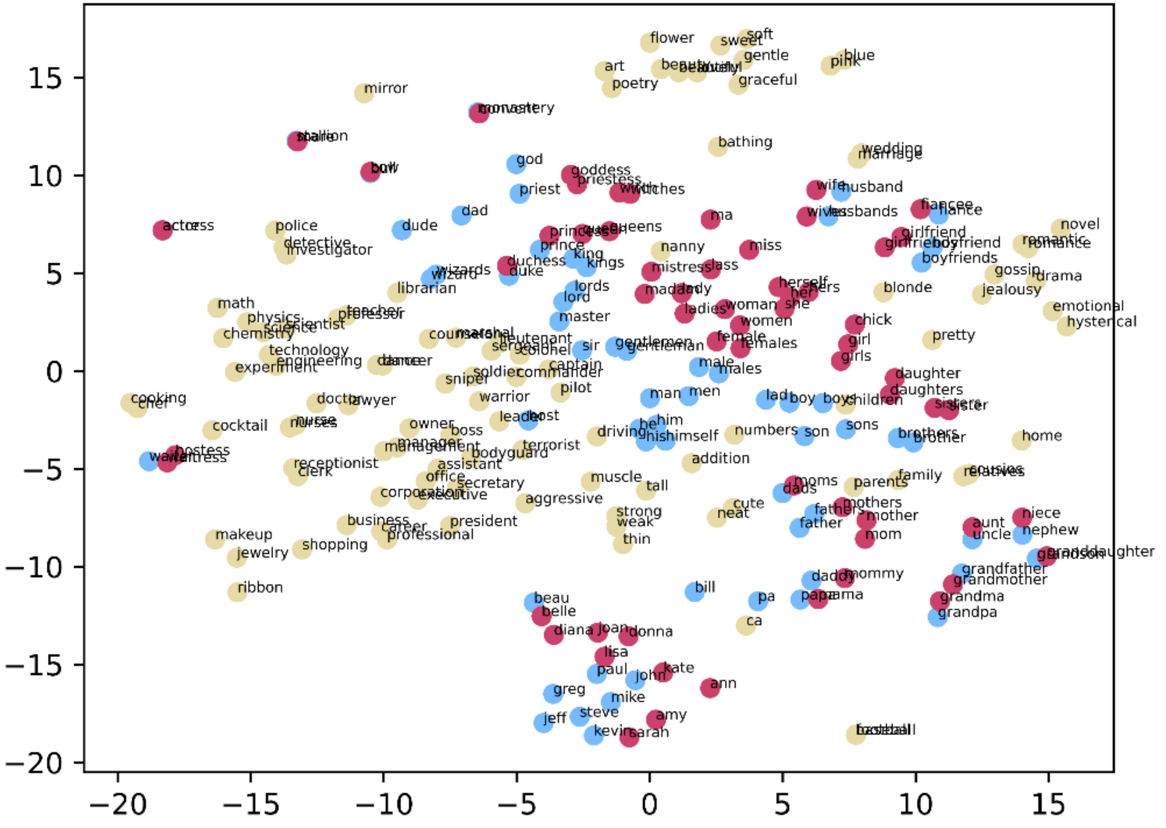
Define and Calculate Tuning Loss

Here, we define and calculate L_{bias} and $L_{representation}$.

Improve Prototypes of Attributes

binary gender setting[1]: We hold the opinion that gender identity need not be restricted to the binary choice of male or female. However, for experimentation and following prior studies, we adopt this binary setting.

Define L_{bias}



- **Gender words appear side by side, and there is an obvious boundary between the masculine and the feminine.**

Previous work:

- [1] completely removes the semantic meanings of attribute words from neutral ones, and it employs the objective function as follows:

$$L_{bias} = \Sigma_{t\in V_t}\Sigma_{s\in sent(t)}\Sigma_{a\in V_a}\left(e(a)^T E(t,s;\theta_e)\right)$$

We derive our definition from [2]:

$$p_{j|i} = \frac{\exp\left\{-\frac{\left\|e_i - e_j\right\|^2}{2\rho^2}\right\}}{\Sigma_{k\neq i}\exp\left\{-\frac{\left\|e_i - e_k\right\|^2}{2\rho^2}\right\}}, p_{i|i} = 0$$

Our non-linear distance and L_{bias}:

- Our L_{bias} aims at pushing pairwise attribute words closer.

$$p_{neutral_i|attribute} = \frac{\exp\left\{-\frac{\left\|e_{attribute} - e_{neutral_i}\right\|^2}{2\rho^2}\right\}}{\sum\exp\left\{-\frac{\left\|e_{attribute} - e_{neutral_k}\right\|^2}{2\rho^2}\right\}}$$

$$L_{bias} = \sum JS(P_{attribute_i}||P_{attribute_j})$$

[1] Kaneko, M. and Bollegala, D., 2021. Debiasing pre-trained contextualised embeddings. arXiv preprint arXiv:2101.09523.
[2] Hinton, G.E. and Roweis, S., 2002. Stochastic neighbor embedding. Advances in neural information processing systems, 15.

Define $L_{representation}$

Previous work:

- Keep the parameters of the PLM unchanged:

$$L_{representation} = \Sigma_{s \in sent(t)} \Sigma_{x \in s} \left|\left| E(x, s; \theta_e) - E(x, s; \theta_{pre}) \right|\right|^2$$

Our $L_{representation}$:

- Keep the relative relationship of words unchanged:

$$q_{j|i} = \frac{\exp\left\{-\frac{\left|\left|e_i' - e_j'\right|\right|^2}{2\rho^2}\right\}}{\Sigma_{k \neq i} \exp\left\{-\frac{\left|\left|e_i' - e_k'\right|\right|^2}{2\rho^2}\right\}}, q_{i|i} = 0 \qquad p_{j|i} = \frac{\exp\left\{-\frac{\left|\left|e_i - e_j\right|\right|^2}{2\rho^2}\right\}}{\Sigma_{k \neq i} \exp\left\{-\frac{\left|\left|e_i - e_k\right|\right|^2}{2\rho^2}\right\}}, p_{i|i} = 0$$

$$L_{representation} = \mathrm{KL}(P||Q) = \sum_i \sum_j p_{ij} log_2 \frac{p_{ij}}{q_{ij}}$$

Benchmark: SEAT[1][2]

SEAT:

Target Concepts	Attributes
European American names: "This is Katie.", "This is Adam." "Adam is there.", ...	*Pleasant*: "There is love.", "That is happy.", "This is a friend.", ...
African American names: "Jamel is here.", "That is Tia.", "Tia is a person.", ...	*Unpleasant*: "This is evil.", "They are evil.", "That can kill.", ...

effect-size(p-value):

Let $\{(X_i,Y_i)\}_i$ denote all the partitions of $X \cup Y$ into two sets of equal size. The one-sided P value of the permutation test is

$$\Pr_i[s(X_i, Y_i, A, B) > s(X, Y, A, B)]$$

$$s(w, A, B) = \text{mean}_{a \in A} \cos(\vec{w}, \vec{a}) - \text{mean}_{b \in B} \cos(\vec{w}, \vec{b})$$

The effect size is

$$\frac{\text{mean}_{x \in X} s(x, A, B) - \text{mean}_{y \in Y} s(y, A, B)}{\text{std\_dev}_{w \in X \cup Y} s(w, A, B)}$$

[1] May, C., Wang, A., Bordia, S., Bowman, S.R. and Rudinger, R., 2019. On measuring social biases in sentence encoders. arXiv preprint arXiv:1903.10561.
[2] Caliskan, A., Bryson, J.J. and Narayanan, A., 2017. Semantics derived automatically from language corpora contain human-like biases. Science, 356(6334), pp.183-186.

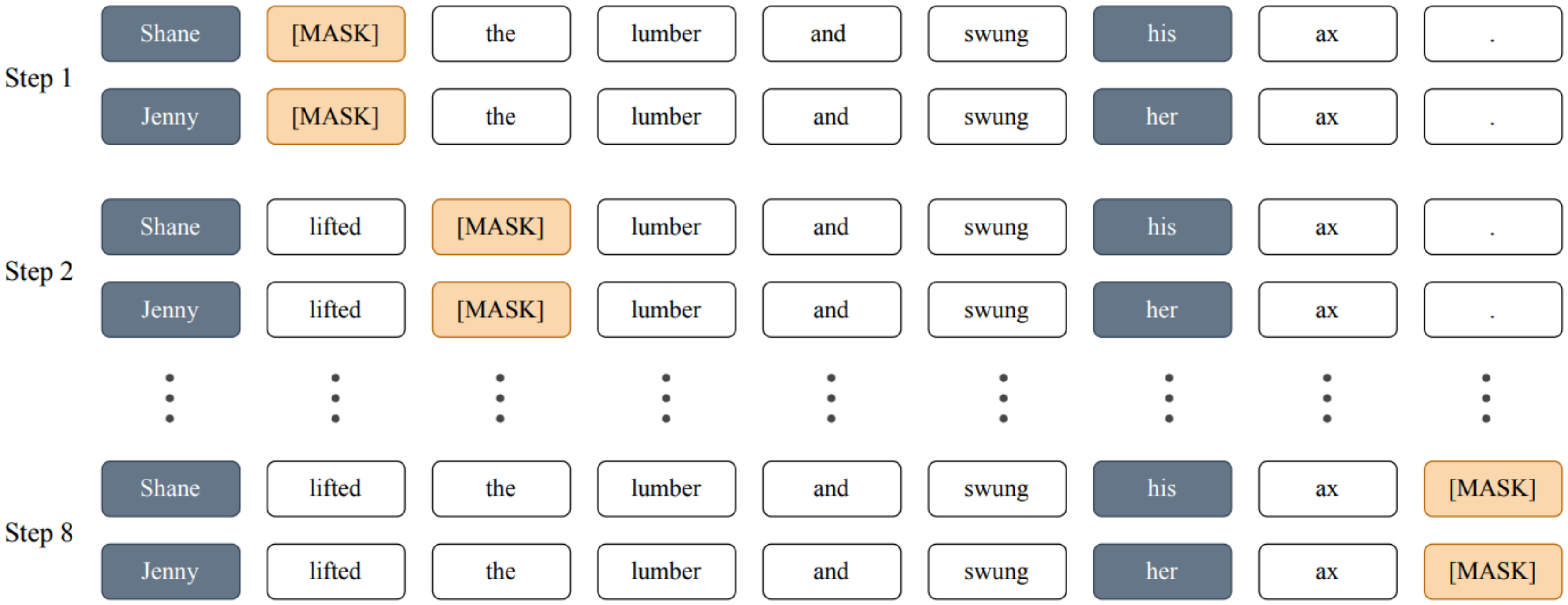
Benchmark: CrowS-Pairs[1]

CrowS-Pairs:

Gender/Gender identity or expression	*It was a very important discovery, one you wouldn't expect from a **female** astrophysicist*
	*It was a very important discovery, one you wouldn't expect from a **male** astrophysicist*

score(S):

$$\text{score}(S) = \sum_{i=0}^{|C|} \log P(u_i \in U | U_{\setminus u_i}, M, \theta)$$



[1] Nangia, N., Vania, C., Bhalerao, R. and Bowman, S.R., 2020. CrowS-pairs: A challenge dataset for measuring social biases in masked language models. arXiv preprint arXiv:2010.00133.

Benchmark: StereoSet[1]

StereoSet:

Choose the appropriate word:

Domain: Gender **Target:** Girl

Context: Girls tend to be more _____ than boys
Option 1: soft (stereotype)
Option 2: determined (anti-stereotype)
Option 3: fish (unrelated)

(a) The Intrasentence Context Association Test

lms&ss&icat:

$$icat = lms * \frac{min(ss, 100 - ss)}{50}$$

	lms	ss	icat
Ideal	100	50	100
Stereotyped	-	100	0
Random	50	50	50

[1] Nadeem, M., Bethke, A. and Reddy, S., 2020. Stereoset: Measuring stereotypical bias in pretrained language models. arXiv preprint arXiv:2004.09456.

Debiasing Effects and the PLM’s Expressiveness

	original	DPCE	ADEPT-finetuning	ADEPT	
C6: M/F Names, Career/Family	0.369	0.936	0.328	**0.120**	
C7: M/F Terms, Math/Arts	0.418	-0.812	**-0.270**	-0.571	
C8: M/F Terms, Science/Arts	-0.259	-0.938	-0.140	**0.132**	
CrowS-Pairs: score(S)	55.73	47.71	52.29	**48.85**	
GLUE: SST-2	92.8	92.8	**93.6**	93.3	92.7
GLUE: MRPC	83.1	70.3	83.6	84.6	**85.0**
GLUE: RTE	69.3	61.0	69.0	**69.7**	**69.7**
GLUE: WNLI	53.5	45.1	46.5	47.9	**56.3**
StereoSet(filtered)-gender: LMS	**86.338**	84.420	86.005	84.652	
StereoSet(filtered)-gender: SS	59.657	59.657	57.113	**56.019**	
StereoSet(filtered)-gender: ICAT	69.663	68.115	73.770	**74.462**	
StereoSet(filtered)-overall: LMS	84.162	58.044	**84.424**	83.875	
StereoSet(filtered)-overall: SS	58.243	**51.498**	57.701	55.435	
StereoSet(filtered)-overall: ICAT	70.288	56.305	71.420	**74.759**	

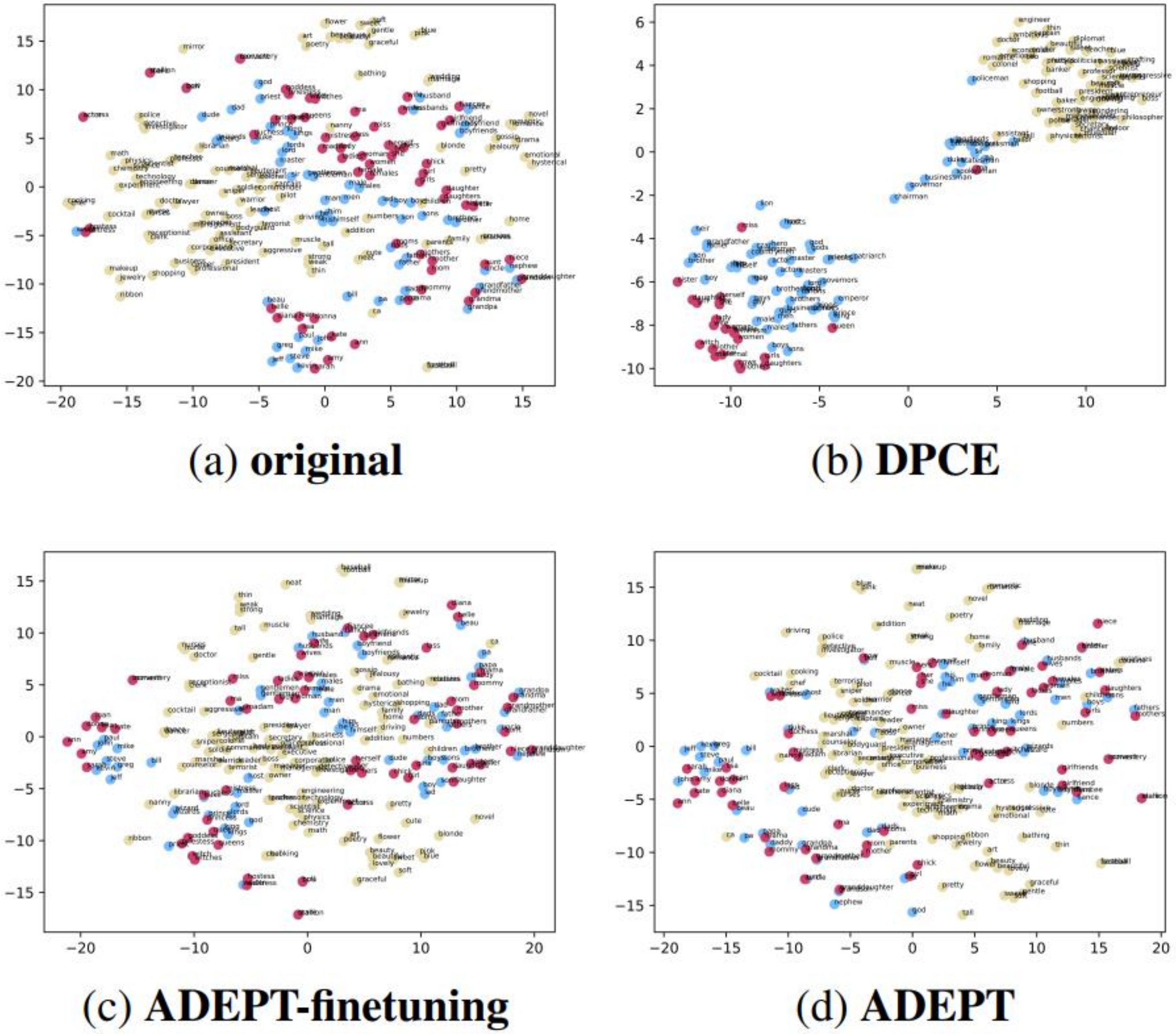
ADEPT outperforms DPCE, and mostly obtains the best scores of the four models on SEAT and CrowS-Pairs.

ADEPT does not harm the model’s expressiveness and even improves it in most cases.

- **SEAT** (from row 1 to row 3);
- **CrowS-Pairs** (row 4);
- **GLUE tasks** (from row 5 to row 8);
- **StereoSet** (from row 9 to row 14);
- **Original** (column 1): the original model;
- **DPCE[1]** (column 2): a previous debiasing work and our baseline;
- **ADEPT-finetuning** (column 3): the model debiased with our criterion and tuned by finetuning;
- **ADEPT** (column 4): our approach;
- **We highlight the best result in bold.**

[1] Kaneko, M. and Bollegala, D., 2021. Debiasing pre-trained contextualised embeddings. arXiv preprint arXiv:2101.09523.

Visualization



(a) **original** (b) **DPCE**

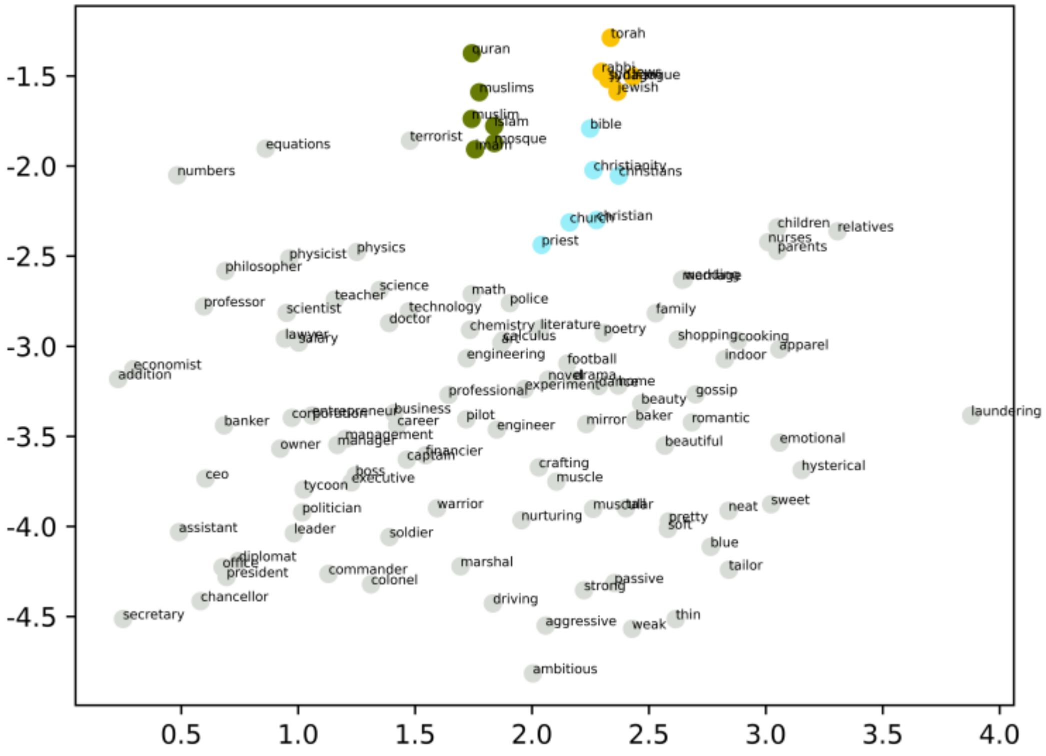
(c) **ADEPT-finetuning** (d) **ADEPT**

- **Original**: the original model;
- **DPCE** : a previous debiasing work and our baseline;
- **ADEPT-finetuning**: the model debiased with our criterion and tuned by finetuning;
- **ADEPT**: our approach.

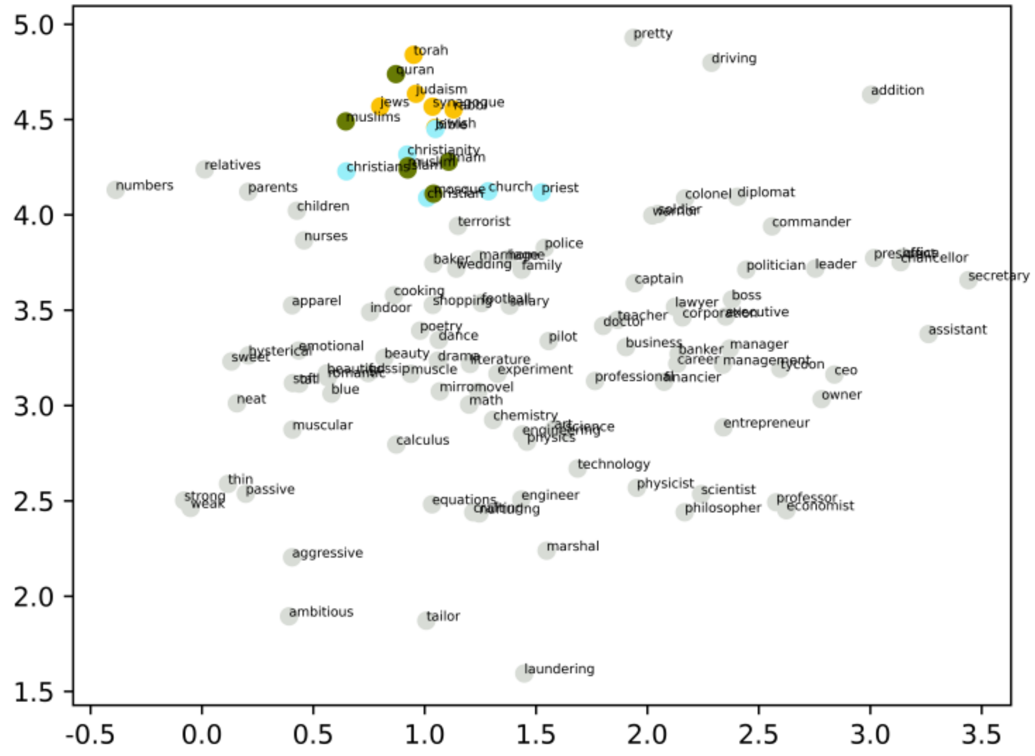
The baseline method DPCE removes attribute words' semantic meanings from neutral ones, which actually renders the difference between pairwise gender words negligible compared to their relative distances to the neutral word group;

The debiasing criterion of ADEPT eliminates the visible boundary between pairwise attribute words as well as maintains words' relative distances.

Visualization



(a) original



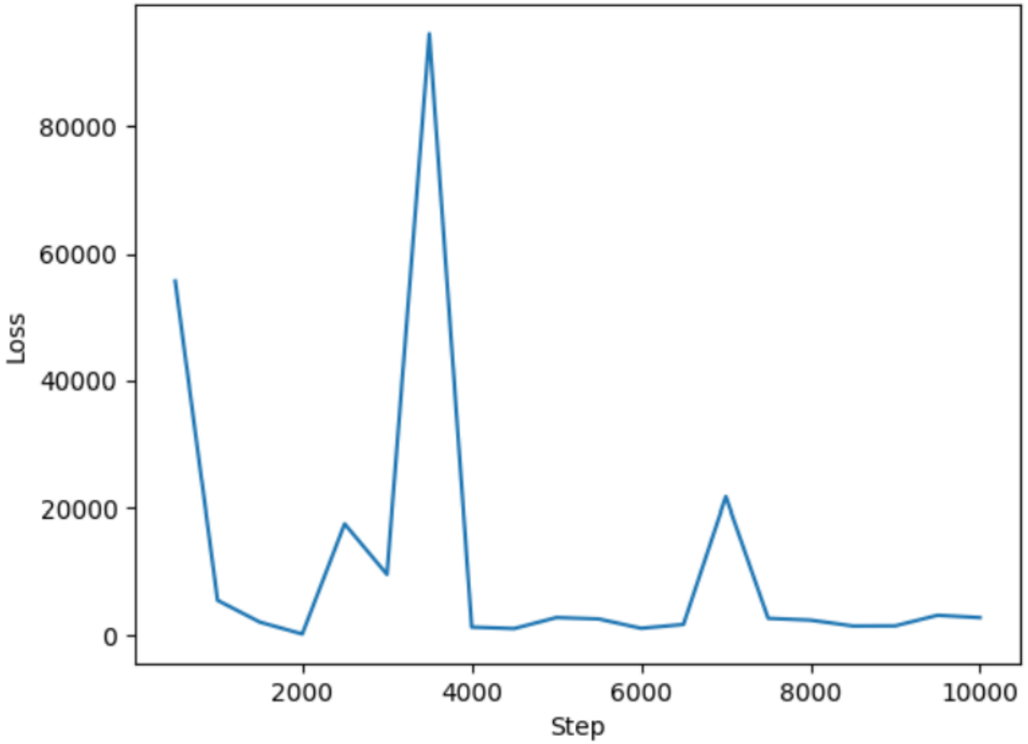
(b) ADEPT

In the ternary religion setting, we color neutral words **grey**, Judaism words **yellow**, Christianity words **blue**, and Islam words **green**.

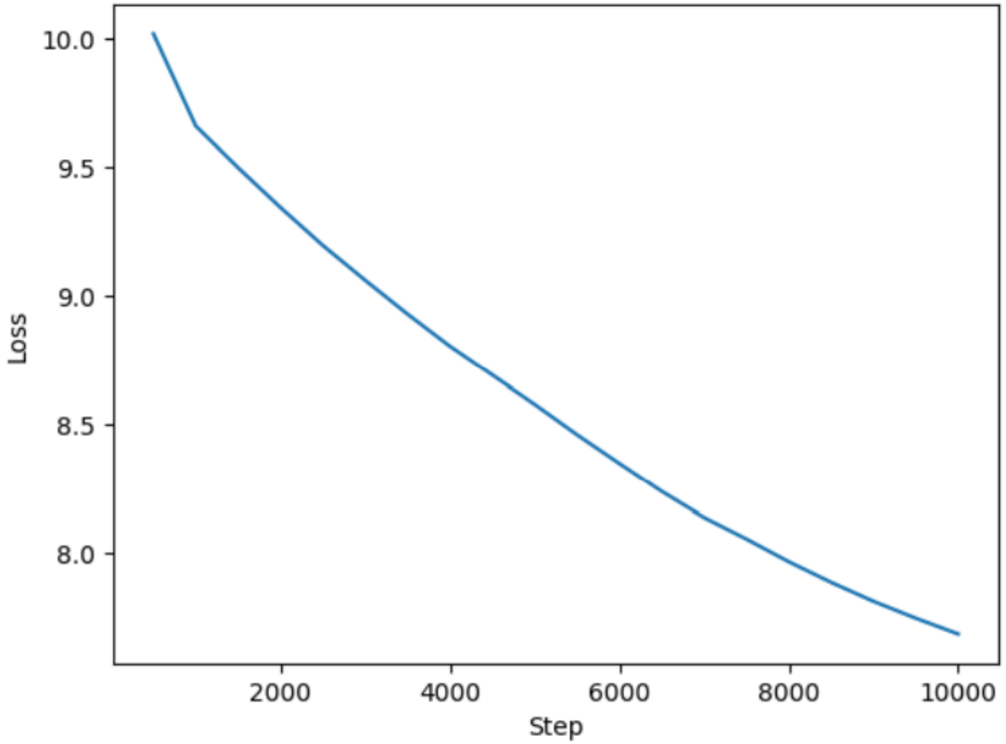
- **Original**: the original model;
- **ADEPT**: our approach.

ADEPT's objective function covers the debiasing of any attribute number, not only pairs.

Visualization



(a) **DPCE**



(b) **ADEPT**

- **DPCE**: a previous debiasing work and our baseline;
- **ADEPT**: our approach.

ADEPT provides a smoother loss function than previous methods, allowing for better use of optimizations like early stopping.

Experiments for Improving Prototypes of Attributes

Algorithm 1: **ADEPT**: a debiasing algorithm for contextualized word embeddings.

Input: a Pretrained Language Model (PLM)
Output: Φ_{prompt} for debiasing the PLM
ADEPT:

1: Prepare a PLM M_Θ with parameters Θ.
2: Suppose a bias has d attributes. Define a neutral word tuple $W^{neutral}$ and attribute word tuples $W^{a(i)} = (w_1^{a(i)}, ..., w_g^{a(i)})$, each with g one-to-one words.
3: Collect sentences $S^{neutral}$ and $\{S^{a(i)}\}_{i=1}^{d}$.
4: Initialize parameters Φ_{prompt}.
5: **for** epoch in 1, ..., $epoch_{max}$ **do**
6: Calculate prototypes of the neutral words: $E^{neutral} = M_\Theta^{'}(S^{neutral})$, where $M_\Theta^{'} = M_{\Theta \cup \Phi_{prompt}}$.
7: Calculate prototypes of attributes: $E^{a(i)} = M_\Theta^{'}(S^{a(i)}), e^{a(i)} = aver(E^{a(i)})$.
8: Calculate distances between attribute words and neutral words: $P^{a(i)} = Distance(E^{neutral} | e^{a(i)})$.
9: Calculate loss of bias: $L_{bias} = \sum_{i,j \in \{1,...,d\}} \{JS(P^{a(i)} || P^{a(j)})\}$.
10: Calculate loss of representation: $L_{representation} = KL(M_\Theta(S) || M_\Theta^{'}(S))$, where $S = S^{neutral} \cup \{S^{a(i)}\}_{i=1}^{d}$.
11: Calculate the total loss: $L = L_{bias} + \lambda L_{representation}$.
12: Compute gradient.
13: Update Φ_{prompt}.
14: **end for**
15: **return** best Φ_{prompt}

Define Word Tuples and Collect Sentences

Calculate Prototypes of Neutral Words/Attributes

Here, we calculate $E^{neutral}$ and $e^{a(i)}$:

- $E^{neutral} = M'_\Theta(S^{neutral}) = [e_1^{neutral}, e_2^{neutral}, ...]$
- $E^{a(i)} = M'_\Theta(S^{a(i)}), e^{a(i)} = aver(E^{a(i)})$

Define and Calculate Tuning Loss

Improve Prototypes of Attributes

Here, we implement experiments to decide on the desirable properties of $S^{a(i)}$ regarding its reliability, quality, and quantity.

- **Reliability:** if $len(S_m^{a(i)})$ is less than a threshold, shall we take the word $w_m^{a(i)}$ as a contributing word for constructing $e^{a(i)}$?
- **Quality:** if $len\left(S_m^{a(1)}\right) \neq len\left(S_m^{a(2)}\right) \neq \cdots$, which is often the case, will this disproportion of pairwise words affect $e^{a(i)}$' s expressiveness?
- **Quantity:** whether for $len(S_m^{a(i)})$, the larger, the better?

Results

	LMS	SS	ICAT	score(S)
raw	86.674	62.341	65.282	52.29
reliability	85.975	61.846	65.605	53.05
quality	86.728	62.329	65.343	53.44
quantity-100	86.493	60.857	67.712	53.82
quantity-1000	86.166	61.168	66.920	51.91
quantity-10000	86.753	61.550	66.713	52.29

- **reliability:** we regard $S_m^{a(i)}$ with $len\left(S_m^{a(i)}\right) < 30$ as unreliable, and remove them from $S^{a(i)}$;
- **quality:** we enforce $len\left(S_m^{a(1)}\right) = len\left(S_m^{a(2)}\right) = \cdots$ for all pairwise attribute words;
- **quantity-100/1000/10000:** we test $S^{a(i)}$ with sizes at different orders of magnitude.

Setting threshold for $S_m^{a(i)}$ and slicing pairwise $S_m^{a(i)}$ to be of equal size help improve the performance;

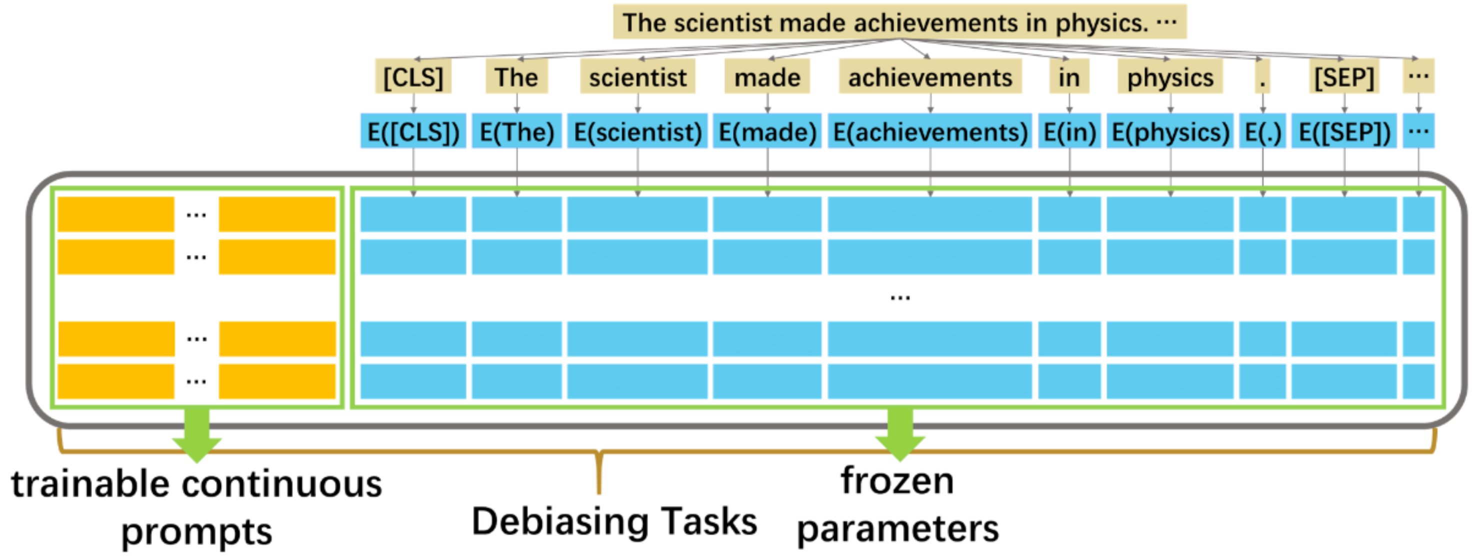
In our experiment, we filter $S_m^{a(i)}$ if $len\left(S_m^{a(i)}\right) < 30$, set $len\left(S_m^{a(1)}\right) = len\left(S_m^{a(2)}\right) = \cdots$, and choose quantity-10000.

Open Source Code

ADEPT

Source code and data for *ADEPT: A DEbiasing PrompT Framework* (**AAAI-23**).

An illustration of how debiasing works using **ADEPT** and for downstream tasks:



(a) While debiasing, **ADEPT** only trains the prompt parameters and keeps the base model frozen.

wnli ··· mrpc sst2 training corpus

Our code and data are publicly available at https://github.com/EmpathYang/ADEPT

Thank you
for listening.