

Better Hit the Nail on the Head than Beat around the Bush: Removing Protected Attributes with a Single Projection

Pantea Haghightakhah[◇] Antske Fokkens^{♣◇} Pia Sommerauer[♣]
Bettina Speckmann[◇] Kevin Verbeek[◇]

[◇]TU Eindhoven, Department of Mathematics and Computer Science
[♣]Vrije Universiteit Amsterdam, Computational Linguistics Text Mining Lab

Abstract

Bias elimination and recent probing studies attempt to remove specific information from embedding spaces. Here it is important to remove as much of the target information as possible, while preserving any other information present. INLP is a popular recent method which removes specific information through iterative nullspace projections. Multiple iterations, however, increase the risk that information other than the target is negatively affected. We introduce two methods that find a single targeted projection: Mean Projection (MP, more efficient) and Tukey Median Projection (TMP, with theoretical guarantees). Our comparison between MP and INLP shows that (1) one MP projection removes linear separability based on the target and (2) MP has less impact on the overall space. Further analysis shows that applying random projections after MP leads to the same overall effects on the embedding space as the multiple projections of INLP. Applying one targeted (MP) projection hence is methodologically cleaner than applying multiple (INLP) projections that introduce random effects.

1 Introduction

Word embedding spaces can contain rich information that is valuable for a wide range of NLP tasks. High quality word embeddings should capture the semantic attributes associated with a word’s meaning. Though we can establish that points representing words with similar semantic attributes tend to be close to each other, it is challenging to reason which attributes are captured to what extent and, in particular, which dimensions capture these attributes (Sommerauer and Fokkens, 2018). But even though we do not (yet) know exactly which attributes are represented in which way, we nevertheless may want to ensure that one particular attribute is no longer present in the embedding. For example, because this attribute is a protected attribute such as gender which can lead to harmful

bias, or because we want to test how the attribute was affecting a system in a probing setup.

There are various approaches to remove a particular attribute from an embedding. Clearly it is important to remove as much of the target attribute as possible while preserving other information present. Arguably linear projections are one of the least intrusive methods to transform an embedding space. Iterative nullspace projection (Ravfogel et al., 2020, INLP) is a recent popular method which follows this paradigm and uses multiple linear projections. INLP achieves *linear guarding* of the protected attribute, that is, a linear classifier can no longer separate instances that have the attribute from instances that do not.

However, it remains unclear to what extent INLP affects the remainder of the embedding space. In general it requires 10-15 iterations (projections) to achieve linear guarding; it is hence likely that other attributes are negatively affected as well. This is particularly problematic in settings where INLP is used to determine what impact a specific attribute has on overall systems.

In this paper, we introduce two methods that find a single targeted projection which achieves linear guarding at once. Specifically, our **Mean Projection [MP]** method uses the mean of the points in each class to determine the projection direction. The mean is more robust to unbalanced class distributions than INLP. The mean can, however, be overly sensitive to outliers. We prove that our second method, the **Tukey Median Projection [TMP]**, finds nearly worst-case optimal projections for any input. Unfortunately, computing the Tukey median (Herbert Edelsbrunner, 1987) is computationally expensive in high dimensions. In our experiments we hence compared only MP to INLP. Specifically, we carried out the gender debiasing experiments of Ravfogel et al. (2020) with both methods. They show that

1. MP only needs one projection for linearly

guarding gender where INLP needs 10-15.

2. MP has less impact on other aspects of the embedding space.

INLP projections improve simlex scores and reduce WEAT scores more than MP. A priori it is unclear why their many projections should have this effect. We investigated and show that the same improvements appear after applying random projections (either after MP or after the first INLP projections). Applying one MP projection to linearly guard an attribute is hence methodologically cleaner than applying multiple (INLP) projections.

2 Related Work

Multiple methods for removing bias from embeddings have been suggested. Bias can be addressed at the level of the training data (Zhao et al., 2018a), the training process (Zhang et al., 2018; Madras et al., 2018), and the resulting model itself (Ravfogel et al., 2020; Bolukbasi et al., 2016; Vargas and Cotterell, 2020). Debiasing an existing model has the clear advantage that it can be done with relatively little data and that the model does not need to be retrained. In this paper, we focus on removing bias through transforming the embedding space.

Transformations of embedding spaces can be performed by means of linear projections (Bolukbasi et al., 2016), multiple linear projections (Ravfogel et al., 2020), or via non-linear kernels (Vargas and Cotterell, 2020). Ravfogel et al. (2020) introduce a method called Iterative Nullspace Projection (INLP) where they attempt to remove the bias by projecting points iteratively. Our methods achieve linear guarding after a single linear projection. As such we directly improve upon the INLP method, which we describe in detail in Section 3. In the remainder of this section, we focus on more recent post-training approaches and applications of INLP.

Similar to Ravfogel et al. (2020), Ravfogel et al. (2022) aim to find a linear subspace of the data such that removing that subspace using projection removes bias in the data optimally. Doing so necessarily requires a metric to measure how well bias has been removed by a specific projection. Ravfogel et al. (2022) use the classification loss of various classifiers as metrics. That is, a higher classification loss after projection corresponds to better bias removal. This directly translates into a minimax optimization problem, which the authors refer to as a minimax game. The authors present

different strategies to solve this optimization problem for their chosen classifiers. A similar strategy was previously described by Haghighatkah et al. (2021, 2022). Here the corresponding minimax optimization problem is referred to as “maximizing inseparability”. Haghighatkah et al. (2022) show that this problem can be solved efficiently under certain convexity assumptions on the metric.

Zhang et al. (2022) independently developed a projection method which is very similar to our MP. Their method also uses class means to find the projection direction, but proceeds via a projection to the origin, and hence needs one additional projection. The goal of the paper, however, is quite different, namely the study of human brain reactions to syntactic and semantic features of words. As such, it does not include a systematic comparison to INLP or a comprehensive analysis of the impact on the embedding space.

Shao et al. (2022) in recent and unpublished work compute linear projection vectors by minimizing the covariance between biased word vectors and the protected attribute. They generally need two projections to achieve comparable or better performance to INLP. The paper does not report on exactly the same experiments as we do, but for all experiments reported in both papers, our method MP performs similarly or better.

Dev et al. (2021a) use a different kind of linear transformation on embedding spaces, namely rotation. Their goal is to disentangle two particular attributes, such as gender and occupation. To do so, they rotate the embedding space in such a way that the subspaces corresponding to the two attributes become orthogonal. On the positive side, their approach does not remove a dimension and hence arguably retains more information in the embedding. However, it will not remove all bias with respect to gender, but only gender bias with respect to a single specified attribute, e.g., occupation. More generally, it will remove the interaction between the two specified attributes and not actually completely remove an attribute from the embedding.

INLP has recently gained importance for probing studies. Probing has been criticized because (1) results are difficult to interpret, and (2) it is usually not possible to test whether the target attribute is also relevant for downstream tasks. Elazar et al. (2021) propose *amnesic probing*, which employs INLP to *remove* the target attribute and then tests change in performance of the adapted embeddings

on downstream tasks. INLP has since been used for this purpose in several other studies (Celikkanat et al., 2020; Nikoulina et al., 2021; Babazhanova et al., 2021; Dankers et al., 2022; Gonen et al., 2022; Lovering and Pavlick, 2022, e.g.). When using a projection-based removal method such as INLP or our MP in a probing setup, it is particularly important to ensure that all other information is preserved. If this is not the case, the probing study may lead to misleading conclusions. We hence recommend future such studies to consider using MP instead of INLP.

3 Projection Methods

In this section we introduce our two new projection methods MP and TMP. Furthermore, we theoretically analyze their relative performance when (linearly) debiasing word embeddings, also in comparison to the existing method INLP. Here we focus on binary classification for ease of explanation. Both methods can however also be used for multi-class classification (see Section 4.1) and then require $n - 1$ projections for n classes. We start with a formal description of the problem we study.

Our input is a set of word embedding vectors in \mathbb{R}^d . We interpret these vectors as a set of points $P = \{p_1, \dots, p_n\}$ in \mathbb{R}^d and often use the term “point” when referring to an embedding vector. Every point (word) has an associated set of discrete attributes ($A = \{a_1, \dots, a_n\}$). Let a^* be the attribute we aim to remove. For simplicity, we assume that a^* is a binary attribute with values $\{-1, +1\}$. We denote the two resulting classes of points as $P^- = \{p_i \in P \mid a_i^* = -1\}$ and $P^+ = \{p_i \in P \mid a_i^* = +1\}$. We want to find a transformation P' of our point set P such that any linear classifier C trained on P' to classify a^* cannot significantly outperform a classifier that labels by the majority class. That is, P' is **linearly guarded** with respect to a^* .

In the following we consider only transformations which consist of one or more projections. Let p be a point in P . We define its projection p_w along a unit vector w as $p_w = p - (p \cdot w)w$. The projection along the vector w maps points to the hyperplane H_w which is orthogonal to w ($H_w \perp w$) and contains the origin.

To evaluate how well INLP, MP, and TMP do in terms of linear guarding, we consider the number of misclassifications with respect to a^* by the best possible linear classifier after a single projection;

the higher the number of misclassifications, the better the method performs. If a single projection is sufficient to achieve linear guarding, then arguably the semantic encoding of other attributes in our word embeddings are preserved as well as possible.

From a theoretical perspective, TMP is the most effective method; it increases the number of misclassifications the most. However, it is costly to compute exactly for data in 300+ dimensions. MP is not as effective in theory as TMP, but it is more effective than INLP. Furthermore, MP is very easy to compute and appears to be very effective in practice. Hence, we evaluate the efficacy of MP extensively in Section 4 and recommend to use MP for linear guarding in practice.

3.1 INLP

Iterative Nullspace Projection (Ravfogel et al., 2020, INLP) debiases by iteratively projecting along a vector computed by a linear classifier. Here we assume that this linear classifier is a linear Support Vector Machine (SVM), as this is the main classifier used in their paper. In that setting, at each iteration, INLP trains an SVM to classify a^* in the point set P . Since a^* is binary, the result of the SVM is a vector w (along with a single real value, which we may ignore). The input points are then projected along w and the resulting point set is used as the input for the next iteration.

If P^- can be separated from P^+ with a hyperplane, then the vector w is the normal of a separating hyperplane and indicates the direction along which there is the largest gap between P^- and P^+ : the SVM margin. If P^- and P^+ cannot be separated by a hyperplane, then the classifier allows misclassifications for a certain penalty, but otherwise still maximizes the margin. In this setting, SVM uses a parameter that controls the trade-off between misclassifications and the margin.

Analysis. The vector w used in the projection essentially maximizes the gap between P^- and P^+ , and hence it is determined only by the points on the “outside” of P^- and P^+ . Specifically, if P^- and P^+ can be separated, then the vector w can be determined solely by the points on the convex hulls¹ of P^- and P^+ , and is not influenced by the points that are interior to the convex hulls. If P^- and P^+ are not separable, then w will be determined by the points on the convex hulls of the correctly classified

¹The convex hull of a set of points is the smallest convex set that contains all of the points.

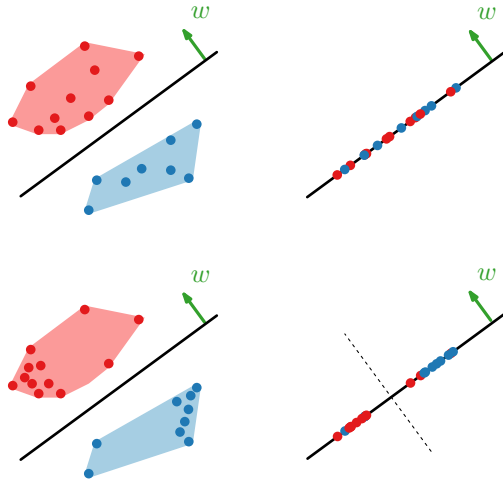


Figure 1: Attribute a^* is represented by color; the resulting point sets after projection along w .

points. Thus, if the number of misclassifications is small, then many of the “more interior” points of P^- and P^+ still do not contribute to determining the projection vector w .

Haghighatkhah et al. (2022) show that the projection along w does eliminate the existence of a perfect linear classifier after projection (that is, there must be at least one misclassification), but this is by far not sufficient for linear guarding. We claim that we must also consider the points interior to the convex hulls of P^- and P^+ , if we wish to obtain an effective projection for linear guarding.

We illustrate this claim with Figure 1, where P^- and P^+ are colored red and blue, respectively. The black line indicates the hyperplane learned by a linear SVM and w is the corresponding vector. The points at the top are distributed roughly evenly in the convex hull of their class (the colored polygon). Hence the projection along w is very effective for linear guarding. The points at the bottom are distributed unevenly in the same convex hulls. The same projection along w is now not very effective: red and blue points can still be separated fairly well by a linear classifier with relatively high accuracy.

We observe that the distribution of all points must be taken into account to compute a projection that is effective for linear guarding. Our two methods (MP and TMP) explore different ways of incorporating the distribution of the interior points when constructing the projection vector w .

3.2 Mean Projection (MP)

Our general goal is to make the distribution of the points in P^- and P^+ as similar as possible after the

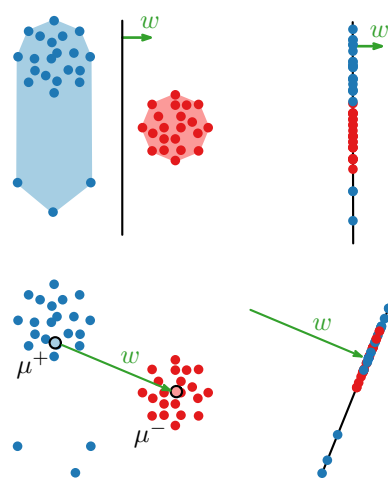


Figure 2: Attribute a^* represented by color; results of INLP (top) and MP (bottom).

projection; that makes it difficult to distinguish the two sets using a linear classifier. One of the most characteristic values of a distribution is its *mean*. Therefore, in the Mean Projection method we ensure that the means of P^- and P^+ are identical after projection. Specifically, let μ^- be the mean of P^- and let μ^+ be the mean of P^+ . We choose the projection vector simply as $w = \mu^+ - \mu^-$. Note that μ^+ and μ^- (and hence w) can easily be computed efficiently by summing up the coordinates of the points in each point set and dividing by the number of points in that set.

Analysis. MP tends to be very effective, since the majority of the points are typically concentrated around their mean. Figure 2 compares INLP (top) and MP (bottom) in a scenario where MP clearly outperforms INLP with respect to linear guarding. However, we can also directly see a weakness of MP: the mean of a point set can be pulled into some direction by (few) outliers. Figure 4 (left) shows another example of this phenomenon. Hence, there exist theoretical instances where MP performs quite poorly with respect to linear guarding. However, we demonstrate in Section 4 that MP is more effective than INLP in practice.

3.3 Tukey Median Projection (TMP)

As illustrated above, the mean can be influenced by outliers. Ideally, we would hence like to identify a “center point” for each point set that has roughly the same number of points in every “direction”. In 1D this center point is known as the *median*. We therefore consider a higher-dimensional version of the median as our center point.

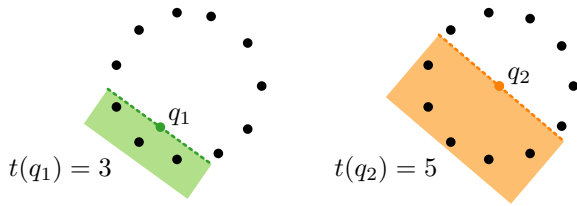


Figure 3: Two points and their Tukey depth.

Let P be a point set in \mathbb{R}^d and let $q \in \mathbb{R}^d$ be another point which is not necessarily in P . The *Tukey depth* $t(q)$ of q with respect to P is the smallest number of points in P in any closed half-space bounded by a hyperplane in \mathbb{R}^d that passes through q . In Figure 3, q_1 is a point with Tukey depth 3 and q_2 is a point with Tukey depth 5. A *Tukey median* τ of a point set P is a point $q \in \mathbb{R}^d$ with the maximum Tukey depth with respect to P among all points in \mathbb{R}^d . Point q_2 in Figure 3 is a Tukey median of the black points.

Now let τ^- be a Tukey median of P^- and let τ^+ be a Tukey median of P^+ . Then the projection vector of the Tukey Median Projection method is simply $w = \tau^+ - \tau^-$.

Analysis. Figure 4 contrasts MP and TMP. The mean of the blue point set is clearly not in the center of the majority of the points due to the outliers. As a result, the projection of MP is not very effective for linear guarding. The Tukey median of the blue point set, however, remains centered near the majority of the points and hence TMP is effective.

One may wonder if there is also an example where TMP performs poorly. We can in fact prove that TMP in general performs quite well for linear guarding. Specifically, we can show that the number of misclassifications after TMP is at least the minimum of the Tukey depths of τ^+ and τ^- . It is known that, for every point set P in \mathbb{R}^d with n points, the Tukey median with respect to P has Tukey depth at least $\left\lceil \frac{n}{d+1} \right\rceil$ (see, e.g., [Herbert](#)

[Edelsbrunner \(1987\)](#)). Finally, we can show that there exist point sets P^+ and P^- such that, after any projection along a single vector, the number of misclassifications is at most $O(n/d)$. Thus, TMP is asymptotically worst-case optimal for maximizing the number of misclassifications after projection. The exact theorem statements and their proofs can be found in [Appendix A](#).

Unfortunately, TMP has one major drawback: computing the Tukey median of a set of points is computationally expensive in high dimensions. Although it is possible to compute an approximate Tukey median efficiently, the Tukey depth of this approximation may be significantly worse than that of the Tukey median (namely, $O(n/d^2)$)². Hence, we generally recommend to use MP in practice.

4 Experiments

In this section, we present our main experimental results.³ We first describe the outcome of the original INLP experiments and how our results compare ([Section 4.1](#)). We then outline the additional experiments we carried out to gain more insight into surprising results of the original INLP experiments as well as those of our own method ([Section 4.2](#)).

4.1 Original Experiments

We compare the performance of INLP and MP on the gender bias experiments reported in [Sections 6.1 and 6.3 in Ravfogel et al. \(2020\)](#). We carried out all experiments with the original (INLP) and our new (MP) method. We could successfully reproduce [Ravfogel et al.’s 2020 results](#). Since the results were nearly identical, we report the INLP results from the original paper. For reasons of space, this section only contains the intuition and main results. We provide a self-contained description

²We are aiming to maximize the number of misclassifications after projection, so $O(n/d)$ is better than $O(n/d^2)$

³The code is available at [tue-alga/debias-mean-projection](#).

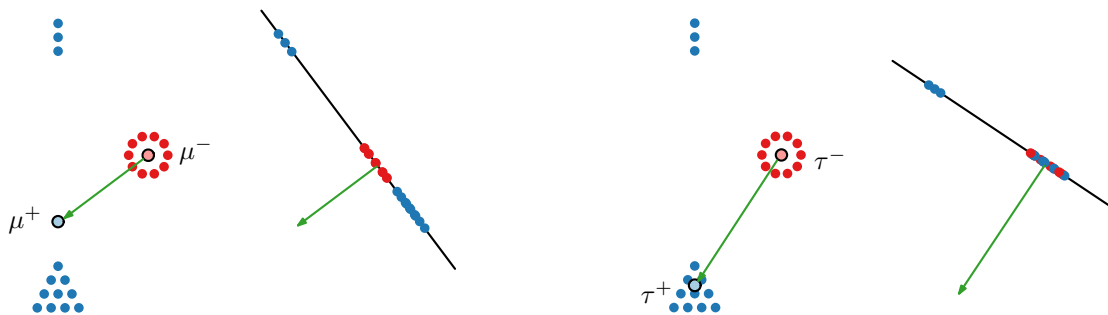


Figure 4: Results of MP (left) and TMP (right).

Experiment	INLP	MP	Interpretation of results
Linear Guarding acc.	34.9 %	34.14 %	Similar
non-linear Classification acc.	85.0%	81.6%	MP slightly better
Effect on Word Neighborhoods	-	-	MP better (more stable neighborhoods)
Embedding Space simplex-999 corr.	0.489	0.373	INLP better (*)
WEAT 6	0.008	0.173	INLP better (*)
WEAT 7	0.084	0.191	INLP better (*)
WEAT 8	-0.310	0.110	MP better
Bias-by-neighbor	73.4%	74.5%	Similar
Gender in the wild BOW acc.	77.1%	76.7%	Similar
Gender in the wild FastText acc.	73.6%	75.6%	Similar
Gender in the wild BERT acc.	74.7%	75.2%	Similar
Gender in the wild BOW TPR-GAP	0.111	0.001	MP better
Gender in the wild FastText TPR-GAP	0.103	0.092	Similar
Gender in the wild BERT TPR-GAP	0.065	0.083%	Similar

Table 1: Summary of the results in Section 4.1: MP generally performs equal to or better than INLP. (*) In Section 4.2 we show that adding random projections to MP achieves the same performance as INLP.

in Appendix B, where all experimental details can be found. We use the same settings and metrics as Ravfogel et al. (2020), unless specified otherwise. Table 1 summarises the results in this section.

General Settings. We use the GloVe word embeddings (Zhao et al., 2018b)⁴ and limit the dataset to the 150,000 most common words. We create the same classes of male, female and neutral embeddings as Ravfogel et al. (2020) following their approach.⁵ A male vector \vec{M} is defined as $\vec{h}e - \vec{s}h$ and a female vector \vec{F} as $\vec{s}h - \vec{h}e$. Our male dataset consists of the 7500 data points closest to \vec{M} and our female dataset to the 7500 data points closest to \vec{F} . We use the same random selection of 7500 neutral words with a cosine similarity of less than 0.3 to \vec{M} as Ravfogel et al. (2020).

We include experiments on two classes (*feminine* and *masculine*) and on three classes (*feminine*, *masculine* and *neutral*). The INLP method uses the datasets to train SVM classifiers as described above. For the MP method, let v_F , v_M and v_N be the mean of respectively feminine, masculine and neutral labeled points of a given set D . Then in experiments on two classes the mean projection method uses the $v_F - v_M$ vector for projection. For experiments with all three classes, we use both $v_N - v_F$ and $v_N - v_M$ vectors for projection.

⁴glove.42B.300d

⁵Gender is not binary. Our experiments are restricted to removing male and female gender bias because we follow the settings from Ravfogel et al. (2020). Our limitations and ethics sections provide more elaborate comments on this matter.

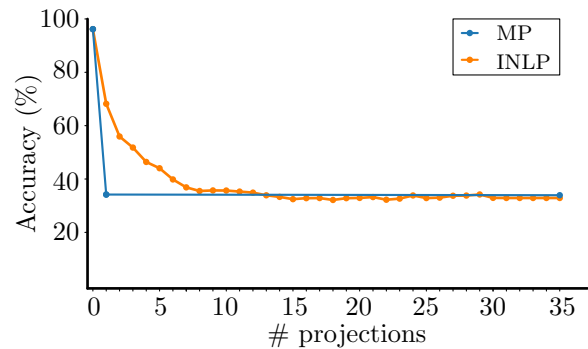


Figure 4: Comparing accuracy of INLP and MP on **three** classes of feminine, masculine and neutral words.

Linear Guarding. We first compare the process of obtaining linear guarding using INLP and MP. Figure 4 illustrates the decrease in accuracy of a linear classifier trained to identify gender. The Mean Projection method brings down the accuracy to 34.18%. It takes INLP 12 iterations to reach a comparable result of 34.9% or at least 6 iterations to reach 39.9% accuracy. Figure 8 in Appendix B shows a similar pattern for binary classification, where MP reaches 50.6% and INLP needs 14 iterations to drop to 50.51%.

Classification. Both methods aim to remove linearly encoded information. Ravfogel et al. (2020) test to what extent the non-linear encoding of gender remains intact by running a 1-hidden-layered MLP with ReLU activation and report 85.0% after 35 iterations with INLP. After a single MP projection, accuracy of the MLP drops to 81.6%.

Effect on the Embedding Space. Ravfogel et al. (2020) show changes in the three nearest neighbors of 40 randomly selected words and 22 gendered given names (see Tables 6 and 7 in Appendix D). MP keeps these environments more stable changing 8 out of 120 neighbors compared to 43 for INLP for random words. MP changes 24 and INLP 56 tokens out of 66 neighbors of the given names. We also observe that 14 of the 24 changes are other given names (5 of which of opposite gender). With INLP, 46 new tokens are not regularly spelled names.

Ravfogel et al. (2020) also investigate how INLP impacts semantic similarity scores on multiple datasets reporting an increase of results, e.g. from 0.373 to 0.489 on simlex-999. Applying MP has less impact with a result of 0.385. This confirms that MP has less impact on the overall space, but also raises the question whether the additional changes caused by INLP’s extra projections is a positive effect. We investigate this in Section 4.2.

WEAT. Caliskan et al. (2017) evaluate bias in embeddings by comparing the distance of known stereotypical terms to terms that are either explicitly male or explicitly female. A higher WEAT score means that stereotypical terms are indeed closer to their stereotyped gender. Like Ravfogel et al. (2020) we use gendered names for representing attributes and take the targets of WEAT 6, 7, and 8. MP achieves WEAT scores of 0.173, 0.191 and 0.110 respectively. The first INLP iteration yields scores of 0.278, 0.347, 0.203 and the full 35 iterations reach 0.008, 0.084 and -0.310 respectively. We investigate this in Section 4.2.

Bias-by-neighbor. Gonen and Goldberg (2019) propose an evaluation that checks whether the 100 nearest neighbors of a term after debiasing carried the same gender bias before debiasing. Ravfogel et al. (2020) show that INLP reduces the gender bias of Bolukbasi et al.’s 2016 set of stereotypical professions from 85.2% to 73.4%. MP reaches a comparable reduction (74.5%).

TPR-GAP: “gender in the wild”. De-Arteaga et al. (2019) propose an approach that measures to what extent debiasing methods can remove gender bias from a system that predicts occupations based on biographies. A fair system should have equal performance for members of different classes and not amplify a bias that is present in the label distribution (Hardt et al., 2016). In the case of gender, the system should perform equally well predict-

ing occupations such as surgeon, caretaker, secretary or marine, regardless of whether the biographical texts are about men or women. De-Arteaga et al. (2019) use the GAP^{TPR} score to measure this. The GAP_{fem} , e.g., quantifies to what extent a given occupation is more probable to be predicted correctly for biographies of women compared to biographies of men. A good debiasing approach should obtain a GAP^{TPR} score that is close to zero while maintaining the classification accuracy that was obtained before debiasing.

Instead of directly debiasing embeddings, we debias representations of entire biographies within the occupation classification system. Like Ravfogel et al. (2020), we use logistic classifiers that take one of three representations of the biographies as input: (1) one-hot BOW, (2) averaged FastText embeddings (Joulin et al., 2017) and (3) the last hidden state of BERT (Devlin et al., 2019). The projections are applied using Pedregosa et al. (2011).

In both approaches, biography representations are debiased on the basis of their gender nullspace for each of the 28 occupations. For the MP setup, we create a mean projection for every occupation o by identifying the projection vector P_o based on the mean of all female and the mean of all male biographies with occupation o . The INLP setup uses logistic regression in 100 iterations for the BOW representations, 150 linear SVM iterations for FastText representations and 300 linear SVM iterations of for the BERT.

Overall, results of both methods are comparable. INLP maintains a higher accuracy on identifying professions for one-hot BOW (77.1% vs 76.7%), where MP has higher accuracy for FastText (75.6% vs 73.6%) and BERT (75.2% vs. 74.7%). For BOW, the GAP score drops by 99.4% when using MP compared to 35% for INLP. Differences for other representations are smaller: MP reduces GAP by 49.5% on FastText compared to INLP’s 43.4%. For the BERT model, INLP outperforms MP reducing the GAP by 64.3% compared to MP’s 52.2%. See Table 5 in Appendix B for the full results.

Summary. In general, we observe that a single MP projection achieves linear guarding where multiple INLP projections are needed and that the rest of the space remains more stable. The improvement of similarity results, as well as INLP’s results on WEAT, raise further questions. Ravfogel et al. (2020) hypothesize that the improved similarity results may be due to a significant gender component

in embeddings that does not correlate with human similarity judgment (Ravfogel et al., 2020, p. 7253). Together with the increased drop in WEAT, this may point to 35 INLP iterations removing more gender information than the single MP projection. This is countered by the result of the non-linear classifier where MP induced a larger decrease than INLP. We investigate the results on similarity and WEAT further in the next subsection.

4.2 Diving Deeper

Simlex-999. We first dive into the impact of INLP on simlex-999 (Hill et al., 2015). Figure 5 illustrates the changes in similarity scores (Correlation) after every iteration of INLP compared to MP (one projection). We observe that the scores mainly increase between the 8th and 14th iteration, remaining relatively stable before and afterwards. Going back to Figure 4, we observe that this increase thus starts when the INLP projections have almost dropped to majority class accuracy, i.e. linear guarding. The main increase in similarity scores thus occurs **after** gender encoding has been removed, which could imply that the increased result is not related to gender. We investigate this by comparing the impact of INLP iterations to iterations that project the dataset along random vectors.⁶

We compare three scenarios: directly applying 35 iterations with random projections to the original model [Random], adding 34 random projection iterations after one MP projection [MP+R] and adding 27 random projection iterations to 8 INLP projections (the point where the score starts to increase) [INLP-8+R]. We carry out 500 runs of this experiment and report the 95% confidence interval of similarity correlation scores for each setting in

⁶Weighted by eigenvalues of all principle components i.e. we choose a random direction from the span of the data.

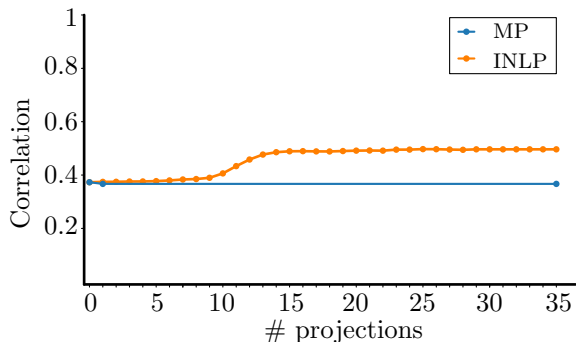


Figure 5: Change in simlex-999 scores for MP and INLP after every iteration.

	95% CI	Mean	Std.
INLP-8+R	[0.47, 0.50]	0.486	0.009
MP+R	[0.46, 0.50]	0.478	0.009
Random	[0.43, 0.46]	0.447	0.009

Table 2: Simlex-999 scores for random projections.

Table 2. Running 35 iterations of INLP increased the semantic similarity score from 0.373 to 0.489. This improvement falls within the confidence interval of MP+R and INLP-8+R. Results of random projections only [Random] do not reach this score, but still clearly improve compared to the original 0.373 with a mean score of 0.447. We thus conclude that the improvement in similarity scores are due to reducing dimensions in general rather than removing (partial) representations of gender.

WEAT. We investigate the WEAT scores in a similar manner. We first inspect the impact on the WEAT score per INLP iteration and then compare this to the impact of projections along weighted random vectors. Figure 6 illustrates the impact of applying INLP for 35 iterations (blue line) and 34 random projection iterations applied after applying MP method (500 runs). Recall that an ideal WEAT score would be close to zero. We observe that MP+R on average ends up with an equal score for WEAT 6, a slightly higher positive (thus worse) score for WEAT 7 and close to zero for WEAT 8 where INLP results in a negative score.

We also compare the effect of applying random projections after 8 INLP iterations. INLP-8+R shows similar results as MP+R. Applying random projections from the start [Random] leads to slightly increasing WEAT scores in most cases (see Figures 9 and 10 in Appendix C). This implies necessity of the first step in removing gender to make a reduction of WEAT by random projections.

We now turn back to what this means for iterative nullspace projections. When inspecting the blue lines in Figure 6, we observe that all WEAT scores initially drop to the same level as with MP after 2-4 iterations and then increase again. Between 20 and 25 iterations we observe a sharp drop, even flipping to a bias in the opposite direction for WEAT 8. These patterns do not reveal a consistent positive impact of applying additional nullspace projections. Ethayarajh et al. (2019) report that results obtained by applying WEAT are brittle and that statistically significant results in opposite directions can be obtained depending on the attribute

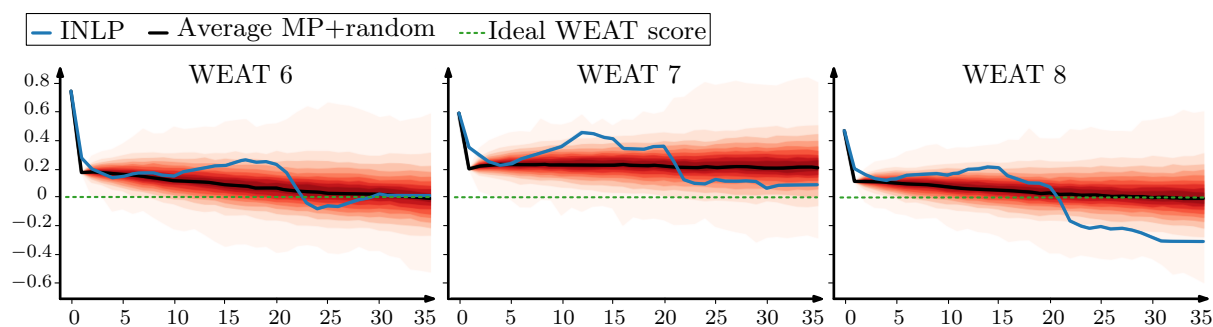


Figure 6: Plot of 500 runs where we first apply MP and then continue with 34 iterations of random projections. The shades of red indicate various levels of confidence (10 levels) in WEAT scores (y -axis) after a certain number of iterations (x -axis): darker red shades reflect WEAT scores occurring with a higher probability.

words selected. They show this in a highly simplified setting (using only one attribute word at the time). Nevertheless, the WEAT sets are relatively small (8 attributes and targets per set) and some of the results we see could be the effect of the specific selection of terms in the set. Overall, this makes the results even more difficult to interpret. We can say, however, that a single MP and the first INLP projections seem beneficial for decreasing WEAT and there is no evidence that continuing iterations of INLP further removes gender bias.

Summary. Our further investigations showed that (1) increases in *simlex-999* scores occur very locally after gender has been removed and that (2) both for similarity and WEAT, random projections after MP or a limited number of INLP iterations lead to similar results as 35 INLP iterations. We therefore conclude that the side effects of continuing to apply INLP are not related to the overall goal of removing encoding of gender, but rather related to the overall effect of dimension reduction.

5 Conclusion

This paper started from the idea that one targeted projection results in similar linear guarding and more stability of the remaining space compared to iterative nullspace projections. We proposed two methods to find such projections: Mean Projection (MP) and Tukey Median Projection (TMP). We compared performance of Mean Projection, which is still susceptible to outliers but more efficient to compute than TMP, to the gender based experiments using 35 Iterative Nullspace Projections (INLP) reported in [Ravfogel et al. \(2020\)](#).

Our results show that MP obtains comparable linear guarding in one projection where INLP requires multiple iterations. They also confirm that

the rest of the space remains more stable through fewer changes in nearest neighbors of randomly chosen words and similarity scores that are closer to the original scores compared to INLP. Two results led to further questions: similarity scores improved after applying INLP and INLP resulted in better WEAT scores than MP. We conducted additional experiments to test whether the extra debiasing iterations performed in INLP end up removing more subtle representations of gender bias that are missed by the single MP iteration.

The results show that it is unlikely that INLP can indeed remove more subtle representations of bias for two reasons: (1) we observed that most effects occur once linear guarding has already been achieved. (2) More importantly, similar performance is obtained by running random projections after either MP or the first 8 INLP iterations. We therefore conclude that these additional effects are not related to removing representations of the targeted attribute, but rather a side effect of reducing the dimensionality of the space. This leads us to the overall conclusion that a single targeted projection is indeed preferable over iterative nullspace projections when aiming to remove an attribute. This finding is particularly important for the recent line of research that uses INLP for interpretability: these studies test the effect of removing a target attribute and their conclusions should not be confounded by other changes to the space.

Our theoretical findings in Section 3 suggest that MP might produce poor results for skewed data. Hence we plan to analyze MP more rigorously on synthetic data, specifically, data where the points are not centered around the mean. In this way we hope to identify scenarios where MP produces poor projections; subsequently we can test if TMP produces better projections for the same scenarios.

6 Limitations

The methods we described and evaluated in this paper have the following limitations.

Dependence on labelled attribute data. The results of all projection methods (INLP, MP and TMP) directly depend on the representativeness of the labelled data points that are used to identify the projections. All methods would suffer from non-representative points. In addition, outliers can hamper the effectiveness of MP.

Binary gender. The binary perspective on gender used in our experiments does not reflect the reality of non-binary gender identities. These experiments focus on gender bias in the form of stereotypical associations with binary gender identities. The underrepresentation of non-binary gender references (e.g. singular *they*, *them* in English) is another form of bias that also affects NLP systems. Recent work has shown that current NLP technologies indeed have difficulties dealing with non-binary gender references (Dev et al., 2021b; Brandl et al., 2022, e.g.). To our knowledge, the question of whether language models also contain stereotypes around non-binary gender has not been addressed yet. If such bias is indeed present, data sparseness is likely to pose a problem for the methods described in this paper.

Suitability for language with grammatical gender is unclear For the application of gender bias, we note that all experiments are run on English. In particular for languages with grammatical gender, it is unclear whether these projections would be able to identify semantic gender bias correctly while avoiding to remove grammatical characteristics of the language: words with the same semantic gender will almost always also have the same grammatical gender. Projections are likely to remove this highly distinctive feature which is also a grammatical property. Its removal may impact downstream applications.

Linear guarding only. The methods described here aim only at (and achieve only) linear guarding. This means that non-linear representations of the attributes remain present in the data (e.g. the one-hidden layered perceptron still achieves an accuracy of 85.0% after 35 INLP iterations and 81.6% after applying MP on binary classification).

7 Ethics

We applied our method to gender debiasing. The approach aims to remove bias which can be seen as helping to address an ethical issue, rather than causing one. Nevertheless, we feel it is important to point out the following. When using the methods discussed in this paper to remove a protected attribute, with the idea to avoid harmful bias, the limitations outlined above **must** be taken into account. This concerns both potential limitations of the effectiveness of the approach and the treatment of gender as a binary variable. Concerning effectiveness, this means considering the fact that (1) non-linear representations are not addressed by these methods and (2) the success of the approach is directly dependent on the quality of data used to identify the projections. Concerning the treatment of non-binary gender, it should be kept in mind that (1) to our knowledge, not much is known yet about stereotypical representations of non-binary gender in language models (2) it is known that current NLP methods experience difficulties in dealing with non-binary gender. In general, debiasing methods must always be carefully evaluated to avoid that users of a system assume the problem is solved, when this is not (completely) the case. Ideally, such evaluation should look beyond the scope of the target. When dealing with gender, this should include looking at what models are doing with non-binary gender references.

In practice, the considerations outlined above mean the following: We do not claim that any of the approaches presented in this paper can fully remove bias from embedding representations. When using debiased embeddings as part of a system, we strongly advise to conduct a critical evaluation with respect to potentially remaining bias using data that are representative of the use-case. This is particularly important if the use-case encompasses the potential for discrimination (e.g. automatic CV analysis).

References

- Madina Babazhanova, Maxat Tezekbayev, and Zhenisbek Assylbekov. 2021. Geometric probing of word vectors. In *29th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, ESANN 2021*, pages 587–592. i6doc. com publication.
- Tolga Bolukbasi, Kai-Wei Chang, James Y Zou, Venkatesh Saligrama, and Adam T Kalai. 2016. [Man](#)

- is to computer programmer as woman is to homemaker? debiasing word embeddings. In *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc.
- Stephanie Brandl, Ruixiang Cui, and Anders Søgaard. 2022. [How conservative are language models? adapting to the introduction of gender-neutral pronouns](#). In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3624–3630, Seattle, United States. Association for Computational Linguistics.
- Aylin Caliskan, Joanna J Bryson, and Arvind Narayanan. 2017. [Semantics derived automatically from language corpora contain human-like biases](#). *Science*, 356(6334):183–186.
- Hande Celikkanat, Sami Virpioja, Jörg Tiedemann, and Marianna Apidianaki. 2020. [Controlling the imprint of passivization and negation in contextualized representations](#). In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 136–148, Online. Association for Computational Linguistics.
- Verna Dankers, Christopher Lucas, and Ivan Titov. 2022. [Can transformer be too compositional? analysing idiom processing in neural machine translation](#). In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3608–3626.
- Maria De-Arteaga, Alexey Romanov, Hanna Wallach, Jennifer Chayes, Christian Borgs, Alexandra Chouldechova, Sahin Geyik, Krishnaram Kenthapadi, and Adam Tauman Kalai. 2019. [Bias in bios: A case study of semantic representation bias in a high-stakes setting](#). In *Proceedings of the Conference on Fairness, Accountability, and Transparency, FAT* '19*, page 120–128. Association for Computing Machinery.
- Sunipa Dev, Tao Li, Jeff M Phillips, and Vivek Srikumar. 2021a. [OSCaR: Orthogonal subspace correction and rectification of biases in word embeddings](#). In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5034–5050, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Sunipa Dev, Masoud Monajatipoor, Anaelia Ovalle, Arjun Subramonian, Jeff Phillips, and Kai-Wei Chang. 2021b. [Harms of gender exclusivity and challenges in non-binary representation in language technologies](#). In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1968–1994.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. [Bert: Pre-training of deep bidirectional transformers for language understanding](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186.
- Yanai Elazar, Shauli Ravfogel, Alon Jacovi, and Yoav Goldberg. 2021. [Amnesic probing: Behavioral explanation with amnesic counterfactuals](#). *Transactions of the Association for Computational Linguistics*, 9:160–175.
- Kawin Ethayarajh, David Duvenaud, and Graeme Hirst. 2019. [Understanding undesirable word embedding associations](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1696–1705, Florence, Italy. Association for Computational Linguistics.
- Hila Gonen and Yoav Goldberg. 2019. [Lipstick on a pig: Debiasing methods cover up systematic gender biases in word embeddings but do not remove them](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 609–614, Minneapolis, Minnesota. Association for Computational Linguistics.
- Hila Gonen, Shauli Ravfogel, and Yoav Goldberg. 2022. [Analyzing gender representation in multilingual models](#). In *Proceedings of the 7th Workshop on Representation Learning for NLP*, pages 67–77.
- Pantea Haghighatkah, Wouter Meulemans, Bettina Speckmann, Jérôme Urhausen, and Kevin Verbeek. 2021. [Obstructing classification via projection](#). In *46th International Symposium on Mathematical Foundations of Computer Science, MFCS 2021*, Leibniz International Proceedings in Informatics, LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik.
- Pantea Haghighatkah, Wouter Meulemans, Bettina Speckmann, Jérôme Urhausen, and Kevin Verbeek. 2022. [Obstructing classification via projection](#). In *Computing in Geometry and Topology Journal*, volume 1.
- Moritz Hardt, Eric Price, and Nati Srebro. 2016. [Equality of opportunity in supervised learning](#). *Advances in neural information processing systems*, 29.
- Herbert Edelsbrunner. 1987. *Algorithms in Combinatorial Geometry*. Springer-Verlag, New York.
- Felix Hill, Roi Reichart, and Anna Korhonen. 2015. [Simlex-999: Evaluating semantic models with \(genuine\) similarity estimation](#). *Computational Linguistics*, 41(4):665–695.
- Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2017. [Bag of tricks for efficient text classification](#). In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 427–431, Valencia, Spain. Association for Computational Linguistics.

- Charles Lovering and Ellie Pavlick. 2022. Unit testing for concepts in neural networks. *arXiv preprint arXiv:2208.10244*.
- David Madras, Elliot Creager, Toniann Pitassi, and Richard Zemel. 2018. [Learning adversarially fair and transferable representations](#). In *International Conference on Machine Learning*, pages 3384–3393.
- Vassilina Nikoulina, Maxat Tezekbayev, Nuradil Kozhakhmet, Madina Babazhanova, Matthias Gallé, and Zhenisbek Assylbekov. 2021. The rediscovery hypothesis: Language models need to meet linguistics. *Journal of Artificial Intelligence Research*, 72:1343–1384.
- Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. 2011. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830.
- Shauli Ravfogel, Yanai Elazar, Hila Gonen, Michael Twiton, and Yoav Goldberg. 2020. [Null it out: Guarding protected attributes by iterative nullspace projection](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 7237–7256. Association for Computational Linguistics.
- Shauli Ravfogel, Michael Twiton, Yoav Goldberg, and Ryan D Cotterell. 2022. [Linear adversarial concept erasure](#). In *Proceedings of the 39th International Conference on Machine Learning (ICML)*, volume 162 of *Proceedings of Machine Learning Research*, pages 18400–18421. PMLR.
- Alexey Romanov, Maria De-Arteaga, Hanna Wallach, Jennifer Chayes, Christian Borgs, Alexandra Chouldechova, Sahin Geyik, Krishnamurthy Kenthapadi, Anna Rumshisky, and Adam Kalai. 2019. [What’s in a name? Reducing bias in bios without access to protected attributes](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4187–4195, Minneapolis, Minnesota. Association for Computational Linguistics.
- Andrew Rosenberg and Julia Hirschberg. 2007. V-measure: A conditional entropy-based external cluster evaluation measure. In *Proceedings of the 2007 joint conference on empirical methods in natural language processing and computational natural language learning (EMNLP-CoNLL)*, pages 410–420.
- Shun Shao, Yftah Ziser, and Shay B. Cohen. 2022. [Gold doesn’t always glitter: Spectral removal of linear and nonlinear guarded attribute information](#). *CoRR*, abs/2203.07893.
- Pia Sommerauer and Antske Fokkens. 2018. Firearms and tigers are dangerous, kitchen knives and zebras are not: Testing whether word embeddings can tell. In *Proceedings of the 2018 EMNLP Workshop Black-boxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 276–286.
- Francisco Vargas and Ryan Cotterell. 2020. [Exploring the linear subspace hypothesis in gender bias mitigation](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2902–2913, Online. Association for Computational Linguistics.
- Brian Hu Zhang, Blake Lemoine, and Margaret Mitchell. 2018. [Mitigating unwanted biases with adversarial learning](#). In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society, AIES ’18*, page 335–340.
- Xiaohan Zhang, Shaonan Wang, Nan Lin, Jiajun Zhang, and Chengqing Zong. 2022. Probing word syntactic representations in the brain by a feature elimination method. In *Proceedings of the AAAI Conference on Artificial Intelligence*.
- Jieyu Zhao, Tianlu Wang, Mark Yatskar, Vicente Ordonez, and Kai-Wei Chang. 2018a. [Gender bias in coreference resolution: Evaluation and debiasing methods](#). In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 15–20. Association for Computational Linguistics.
- Jieyu Zhao, Yichao Zhou, Zeyu Li, Wei Wang, and Kai-Wei Chang. 2018b. [Learning gender-neutral word embeddings](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4847–4853. Association for Computational Linguistics.

A Tukey Median Projection

Theorem 1. *Let q^- be a point with Tukey depth $t(q^-)$ with respect to P^- and let q^+ be a point with Tukey depth $t(q^+)$ with respect to P^+ . Then the projection of P along vector $w = q^+ - q^-$ ensures that any linear classifier must make at least $\min(t(q^-), t(q^+))$ misclassifications after projection.*

Proof. Let τ be the Tukey medians of P^- and P^+ after projection (by construction, these Tukey medians have been projected onto the same point). Now let H be the hyperplane corresponding to some linear classifier after projection. Let H_{\leftrightarrow} be the hyperplane obtained by shifting H to contain the point τ . Let H_{\leftrightarrow}^- be the halfspace bounded by H_{\leftrightarrow} that contains the majority of P^- after projection, and let H_{\leftrightarrow}^+ be the other halfspace bounded by H_{\leftrightarrow} . By construction of τ , there must be at least $t(q^-)$ points of P^- in H_{\leftrightarrow}^+ and at least $t(q^+)$ points of P^+ in H_{\leftrightarrow}^- , which are misclassified by the linear classifier defined by H_{\leftrightarrow} . By shifting the hyperplane back from H_{\leftrightarrow} to H , we can see that either the misclassified points of P^- or the misclassified points of P^+ remain misclassified by the linear classifier defined by H . Thus, any linear classifier must misclassify at least $\min(t(q^-), t(q^+))$ points after projection by TMP. \square

We can show that TMP is asymptotically worst-case optimal in minimizing the number of misclassifications after projection. For that we need the following technical lemma.

Lemma 1. *Let Δ_d be the regular d -dimensional simplex embedded in \mathbb{R}^d such that the center of Δ_d lies at the origin and the $d + 1$ vertices of Δ_d , represented by the vectors v_1, \dots, v_{d+1} , are at unit distance from the origin. For any point $p \in \mathbb{R}^d$, there exist a unit vector r such that $(v_i \cdot r) \leq (p \cdot r) - \frac{1}{d}$ for at least d vertices of Δ_d .*

Proof. As the vectors v_1, \dots, v_{d+1} span the entirety of \mathbb{R}^d , we can write p as an affine combination $p = \sum_{i=1}^{d+1} \alpha_i v_i$, where $\sum_{i=1}^{d+1} \alpha_i = 1$. Without loss of generality, let α_j be the largest coefficient among all α_i for $1 \leq i \leq d + 1$. We pick $r = v_j$. Since Δ_d is a regular simplex, it is known that $(v_i \cdot v_j) = \frac{-1}{d}$ for $v_i \neq v_j$. Using the linearity

of the dot product, we now get the following:

$$\begin{aligned} (p \cdot r) &= \sum_{i=1}^{d+1} \alpha_i (v_i \cdot r) \\ &= \alpha_j - \frac{1}{d} \sum_{1 \leq i \leq d+1, i \neq j} \alpha_i \\ &\geq \alpha_j - \frac{1}{d} \sum_{1 \leq i \leq d+1, i \neq j} \alpha_j \\ &= \alpha_j \left(1 - \frac{d}{d}\right) \\ &= 0. \end{aligned}$$

On the other hand we have that $(v_i \cdot r) = \frac{-1}{d}$ for all $v_i \neq v_j$, and hence the stated inequality holds for all $v_i \neq v_j$. \square

We can now prove our main result.

Theorem 2. *For any $d > 0$ and $n \geq m > 0$, there exist point sets P of m points and Q of n points in \mathbb{R}^{d+1} , such that for any projection of P and Q along a unit vector w , there exists a hyperplane H in \mathbb{R}^{d+1} with at least $\lfloor \frac{md}{d+1} \rfloor$ points of P on one side of H and all points of Q on the other side of H after projection.*

Proof. Let e_1, \dots, e_{d+1} be the basis vectors spanning \mathbb{R}^{d+1} . We first prove the statement for the case that $m = d + 1$. Specifically, let P be formed by the vertices of the regular d -dimensional simplex (as in Lemma 1) centered at the origin and restricted to the d -dimensional subspace spanned by the first d basis vectors e_1, \dots, e_d . Furthermore, let Q be an arbitrary set of n points such that all points are within Euclidean distance less than ε of the center point $q = (0, \dots, C)$ (for some $C > 0$ and $\varepsilon > 0$) in \mathbb{R}^{d+1} .

Let w be any unit projection vector. We write w as $w = \alpha e_{d+1} + \beta z$, where z is a unit vector in the subspace spanned by e_1, \dots, e_d (such that we have $\alpha^2 + \beta^2 = 1$). Let q_w be the center of Q after projection along w . It is easy to see that P and Q are linearly separable after projection if $\|q_w\| \geq 1 + \varepsilon$ (the center of P remains at the origin after projection). We have that:

$$\begin{aligned} q_w &= q - (w \cdot q)w \\ &= C e_{d+1} - \alpha C (\alpha e_{d+1} + \beta z) \\ &= e_{d+1} C (1 - \alpha^2) - \alpha \beta C z. \end{aligned}$$

Since z is independent from e_{d+1} we get that $\|q_w\| \geq C(1 - \alpha^2) = C\beta^2$, and hence $\|q_w\| \geq$

$1 + \varepsilon$ if $\beta^2 \geq \frac{1+\varepsilon}{C}$. We may therefore assume that $\beta^2 < \frac{1+\varepsilon}{C}$.

Now let x be the d -dimensional point obtained by omitting the $(d+1)^{\text{st}}$ coordinate of q_w . We apply Lemma 1 to P with $p = x$ to obtain a corresponding unit vector r . We extend r with an additional coordinate with the value 0 to obtain the $(d+1)$ -dimensional vector $R = (r, 0)$. We then define the hyperplane H as the hyperplane that satisfies $(p \cdot R) = (q_w \cdot R) - \varepsilon$ for points $p \in \mathbb{R}^{d+1}$.

We now show that H has the desired properties. Let u_i be one of the d points of P such that $(u_i \cdot r) \leq (x \cdot r) - \frac{1}{d}$, and let u'_i be the point obtained by projecting u_i along w . We get that:

$$\begin{aligned} u'_i &= u_i - (w \cdot u_i)w \\ &= u_i - \beta(z \cdot u_i)w \\ &= u_i - \beta'_i w, \end{aligned}$$

where $\beta'_i \in [-\beta, \beta]$. By construction of x , we also get that $(x \cdot r) = (q_w \cdot R)$. We then get the following:

$$\begin{aligned} (u'_i \cdot R) &= (u_i \cdot R) - \beta'_i(w \cdot R) \\ &= (u_i \cdot r) - \beta'_i \beta(z \cdot R) \\ &\leq (u_i \cdot r) + \beta^2 \\ &\leq (x \cdot r) + \beta^2 - \frac{1}{d} \\ &= (q_w \cdot R) + \beta^2 - \frac{1}{d}. \end{aligned}$$

Thus, if $\beta^2 - \frac{1}{d} < -\varepsilon$, then $(u'_i \cdot R) < (q_w \cdot R) - \varepsilon$. Since $\beta^2 < \frac{1+\varepsilon}{C}$, we can choose $C = 4d$ and $\varepsilon = \frac{1}{2d}$ to ensure this property, as then $\frac{1+\varepsilon}{C} - \frac{1}{d} < \frac{1}{2d} - \frac{1}{d} = -\varepsilon$.

Now let v_j be one of the points of Q , and let v'_j be the point obtained by projecting v_j along w . Since the projection along w can only shrink distances, we get that $\|v'_j - q_w\| \leq \|v_j - q\| < \varepsilon$. This implies that $((v'_j - q_w) \cdot R) > -\varepsilon$, which can be rewritten as $(v'_j \cdot R) > (q_w \cdot R) - \varepsilon$. As a result, d points of P are on one side of H and n points of Q are on the other side of H , as required.

If P contains $m \neq d+1$ points, then we can simply follow the construction above and distribute the points arbitrarily close to the vertices of the regular simplex, such that there are at most $\left\lceil \frac{m}{d+1} \right\rceil$ points around each vertex. As a result, n points of Q will be on one side of H and at least $m - \left\lceil \frac{m}{d+1} \right\rceil = \left\lfloor \frac{md}{d+1} \right\rfloor$ points of P will be on the other side of H , which completes the proof. \square

Corollary 1. For any $d > 1$ and $n \geq m > 0$, there exists a set of points P in \mathbb{R}^d with $|P^-| = m$ and $|P^+| = n$ such that, for any projection of P along a single unit vector w , the number of misclassifications of the best possible linear classifier after projection is at most $\left\lceil \frac{m}{d} \right\rceil$.

Proof. Choose P and the binary attribute a^* such that P^- and P^+ match P and Q in the statement of Theorem 2 (note that the theorem has to be applied with dimension $d-1$). Then, for any projection of P along a single vector, there exists a hyperplane H such that $\left\lfloor \frac{m(d-1)}{d} \right\rfloor$ points of P^- are on one side of H and all points of P^+ are on the other side of H . Thus, the linear classifier defined by H misclassifies at most $\left\lceil \frac{m}{d} \right\rceil$ points, and hence the best possible linear classifier must perform at least as well. \square

We now return to the number of misclassifications caused by TMP. It is known that, for every point set P in \mathbb{R}^d with n points, the Tukey median with respect to P has Tukey depth at least $\left\lfloor \frac{n}{d+1} \right\rfloor$. As a result, by using the projection of TMP, any linear classifier must make at least $\left\lfloor \frac{m}{d+1} \right\rfloor$ misclassifications after projection by Theorem 1, where m is the size of the smallest set among P^- and P^+ . In the worst-case, this nearly matches the upper bound of $\left\lceil \frac{m}{d} \right\rceil$ according to Corollary 1.

B Experiments

For convenience, we provide a detailed self-contained description of the experiments here. The first paragraph is an exact repetition (included here for convenience). All others are extended versions of the one included in the main text. This version can be read either instead of or next to the shorter version in the main text.

General Settings. We use the GloVe word embeddings (Zhao et al., 2018b)⁷ and limit the dataset to the 150,000 most common words. We create the same classes of male, female and neutral embeddings as Ravfogel et al. (2020) following their approach. A male vector \vec{M} is defined as $\vec{h}e - \vec{s}he$ and a female vector \vec{F} as $\vec{s}he - \vec{h}e$. Our male dataset consists of the 7500 data points closest to \vec{M} and our female dataset to the 7500 data points closest to \vec{F} . For the neutral dataset we use Ravfogel et al.’s 2020 seed to obtain the same random

⁷glove.42B.300d

selection of 7500 words with a cosine similarity of less than 0.3 to \vec{M} .

We include experiments on two classes (*feminine* and *masculine*) and on three classes (*feminine*, *masculine* and *neutral*). The INLP method uses the datasets to train SVM classifiers as described above. For the MP method, let \vec{v}_F , \vec{v}_M and \vec{v}_N be the mean of respectively feminine, masculine and neutral labeled points of a given set D . Then in experiments on two classes the mean projection method uses the $\vec{v}_F - \vec{v}_M$ vector for projection. For experiments with all three classes, we use both $\vec{v}_N - \vec{v}_F$ and $\vec{v}_N - \vec{v}_M$ vectors for projection.

Linear Guarding. The first set of experiments illustrates the process of gender information being removed from embeddings using each method. We apply the INLP and MP algorithms to the data labeled feminine, masculine and neutral described above and investigate to what extent a linear classifier can still identify the original gender of the word. The results of this classifier should decrease as gender information is removed from the set. For INLP, we use a L2-regularized SVM with the same parameters and random seeds as Ravfogel et al. (2020). We can observe changes over the course of performing multiple projections for INLP. For MP, only a single projection is needed.

We test the classifier on the same train, test and development set as Ravfogel et al. (2020). The dataset contains three classes (feminine, masculine and neutral). Like Ravfogel et al. (2020), we report the results for all three classes as well as two classes (male and female). Since all classes are equally distributed, we report on the classifiers accuracy. A set of completely debiased embeddings should yield a result that is equal to chance.

Figure 7 presents the results of 35 iterations of INLP and the single iteration of MP on three classes of the data. Before applying projections, a linear classifier achieves perfect classification. The mean projection method brings down the accuracy to 34.18%. It takes INLP 12 iterations to reach a comparable result of 34.9% or at least 6 iterations to reach 39.9% accuracy. We repeat the experiment on two classes (masculine and feminine) and provide the results in Figure 8. Here after applying the MP method, an accuracy of 50.6% is reached while INLP needs 14 iterations to get down to 50.51% accuracy. These results show that with MP, one projection is enough whereas multiple projections are needed to achieve a similar result with INLP.

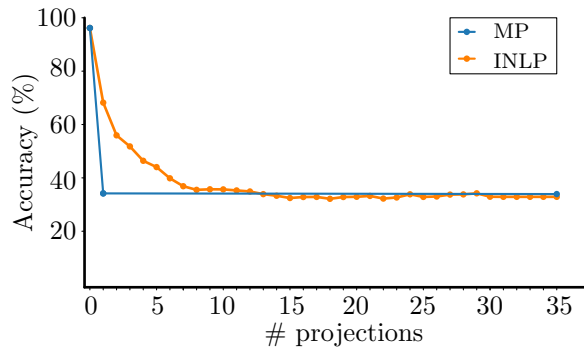


Figure 7: Comparing accuracy of INLP and MP on **three** classes of feminine, masculine and neutral words.

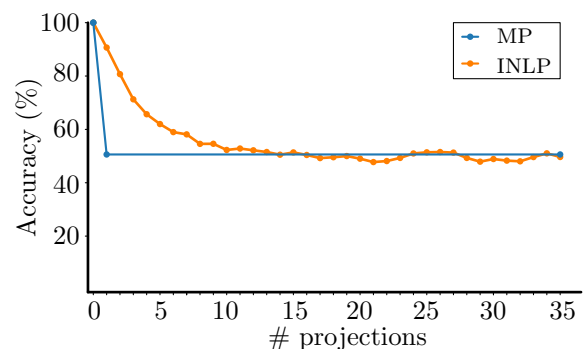


Figure 8: Comparing accuracy of INLP with MP on **two** classes of feminine and masculine words.

The results of INLP stabilizes at approximately the same result as MP’s single projection.

Classification. Both methods only aim to remove linearly encoded information. We compare to what extent they leave non-linear encoding of gender in tact by running a 1-hidden-layered MLP with ReLU activation. This MLP manages to identify gender with 85.0% after 35 iterations. After a single MP projection, accuracy of the MLP drops to 81.6%. Overall these results confirm that MP needs a single projection to achieve equal linear guarding as multiple INLP projections and a comparable (slightly better) result on the remaining non-linear encoding.

Clustering. Another way of assessing the debiasing methods is to test to what extent the embeddings can still form clusters representative of the male and female class. If the bias has been removed successfully, the clusters should no longer reflect gender bias information. We use the V-measure (Rosenberg and Hirschberg, 2007) to assess to what extent the two classes are intertwined. This measure is the harmonic mean of two components: *homogeneity* (reflecting how many points in a cluster

belong to the same class) and *completeness* (reflecting how many points of a class ended up in the same cluster). The homogeneity component assesses to what extent samples of a single cluster have the same class i.e. are similar. Completeness on the other hand measures to what extent data points with the same labels are in the same cluster. A perfect V measure (score 1.0) indicates perfect clustering with respect to the gold labels, where equal distribution of class members from all classes over all clusters yields a score of 0. We apply K -means clustering with $K = 2$ to the male and female gendered words in the test split, then we calculate the V-measure.

We use the 2000 most biased female and male words as [Ravfogel et al. \(2020\)](#) suggests. Using the MP method reduces the V-measure to 0.53% after a single projection. The V-measure reaches 0.67% after applying the INLP method with all 35 iterations on the this set.

Exploring Neighbors. We inspect changes in the direct environment of 40 words randomly selected by [Ravfogel et al. \(2020\)](#) as well as their selection of 22 gendered given names. If a debiasing method is targeted, it should not affect the neighbors of the randomly selected words and only lead to changes in the neighborhood of the words that should be affected by debiasing (in this case the 22 gendered given names, where new name of the opposite gender may appear).

When applying MP, only 8 out of 120 neighbors of random words changed, compared to 43 after changes caused by INLP. For given names, MP yielded 24 changes out of 66 compared to 56 for INLP. Overall, it seems that INLP leads to more changes in the neighborhoods of words targeted by debiasing as well as other words.

Upon manual inspection, we observed that MP mainly leads to names being replaced by other given names in the neighborhood of gendered names (14 times, 5 times a name of opposite gender). INLP’s new contexts contain more common nouns and rare non-given names (46 tokens that end up in the immediate context of a gendered name are not regularly spelled given names). Another noticeable difference is that MP appears to have moved the name *Ariel* to its interpretation of a location, whereas it ended up near other terms related to Disney after applying INLP. The full results can be found in Tables D and 7 in Appendix D.

Similarity Experiments Word debiasing methods should only affect information that reflects the bias and leave the rest of the space unchanged. We investigate the effects of both debiasing methods on three word similarity datasets, namely, simlex-999, Sim353 (both relatedness and similarity dataset) and Mturk-771. The datasets can serve as a reflection of how the debiasing methods affect the semantic space. While it is possible that the similarity datasets contain gendered words, it can be assumed that the majority of word pairs does not have an explicit gendered meaning. We verify this assumption by means of manual analysis of the simlex-999 pairs and found 62 pairs where one or both terms had a meaning that explicitly contains gender. Our manual annotations can be found in our Github repository. We report the Spearman correlation between the similarity scores given by the annotators and the cosine distance before and after debiasing with INLP and MP in Table 3.

Dataset	Original	MP	INLP
simlex-999	0.373	0.385	0.489
WordSim353 - Sim	0.695	0.690	0.799
WordSim353 - Rel	0.599	0.589	0.698
Mturk-771	0.684	0.691	0.728

Table 3: The correlation of the cosine distances with the similarity scores in the Original space, after MP debiasing and after INLP debiasing.

The results on these similarity tests remain relatively stable after applying MP, whereas we see a clear increase in correlation scores after INLP is applied. This result confirms that the overall embeddings change less when applying MP compared to INLP. The observed improvement caused by INLP indicate that more research is needed to determine how INLP affects the entire semantic space. If iterative projections can improve the overall quality, the additional effect they have on the space may actually be desirable. In particular, [Ravfogel et al. \(2020\)](#) hypothesize that this increase may be due to words in the datasets containing a gender component which is not perceived by humans, but do not investigate the matter further. We further explore this possibility in our additional experiments described in Section 4.2 (see main content above).

WEAT. [Caliskan et al. \(2017\)](#) propose the Word Embedding Association Test (WEAT) that can measure bias inferred by the semantic similarity between groups of words. We have two groups of

target words for instance $T_1 = \{ \text{programmer, professor, engineer, ...} \}$, $T_2 = \{ \text{nurse, teacher, librarian, ...} \}$ and two set of attribute words $A_1 = \{ \text{man, male, masculine, ...} \}$, $A_2 = \{ \text{woman, female, feminine, ...} \}$. If the word embeddings representing these target words are not biased, then the relative similarities between the target words (T_1, T_2) and attributes (A_1, A_2) should be equivalent. In other words: words such as *secretary* and *programmer* should not be more similar to either of the two groups of gendered attributes.

WEAT measures the association between the target groups and attributes: a biased representation is expected to maintain this association (reflected by a high score), whereas WEAT scores on representations with low to no bias should be close to zero. A WEAT score of 0 indicates that words from the target groups are, on average, equally strongly associated with A_1 and A_2 . Like Ravfogel et al. (2020), we follow Gonen and Goldberg (2019) and represent the attributes through gendered names rather than attributes. We then take the target words of the same the three tests used by Ravfogel et al. (2020) (*WEAT 6*, *WEAT 7* and *WEAT 8*) in Table 4.⁸ *WEAT (6)* includes targets words referring to career and family attributes, *WEAT (7)* has target words related to math and art and *WEAT (8)* has science and art target words.

Both MP and INLP reduce the WEAT scores. For *WEAT 6* and *7*, 35 INLP projections lead to the largest reduction. For *WEAT 8*, MP provides the best results followed by a single INLP iteration (rather than the full 35). The negative value of *WEAT 8* indicates a stronger bias in opposite direction that the remaining bias present after the first INLP iteration or after the MP projection. This leads us to wonder how to interpret these results. We therefore investigate them further in Section 4.2.

Bias-by-neighbor. Gonen and Goldberg (2019) show that even if a debiasing method manages to remove a biased vector from explicitly gender words, it may still be close to other words with the same stereotypical connotation (e.g. *nurse* remains close to *receptionist* and *teacher*). The **bias-by-neighbors** measure captures such remaining bias by providing the percentage of the 100 closest

⁸Ravfogel et al. (2020) report p-values of associations rather than the values. We present the values instead, because the p-values do not distinguish between positive and negative WEAT scores.

Original		
<i>WEAT 6</i>	<i>WEAT 7</i>	<i>WEAT 8</i>
0.738	0.584	0.467
INLP-35		
<i>WEAT 6</i>	<i>WEAT 7</i>	<i>WEAT 8</i>
0.008 (-98%)	0.084 (-86%)	-0.310 (-34%)
INLP-1		
<i>WEAT 6</i>	<i>WEAT 7</i>	<i>WEAT 8</i>
0.278 (-62%)	0.347 (-41%)	0.203 (-56%)
MP		
<i>WEAT 6</i>	<i>WEAT 7</i>	<i>WEAT 8</i>
0.173 (-76%)	0.191 (-66%)	0.110 (-76%)

Table 4: WEAT scores of tests *WEAT 6*, *WEAT 7*, and *WEAT 8* of the original embeddings, after INLP debiasing with 35 iterations and after MP debiasing. For each score, you can see the decrease from the original WEAT scores.

words that were male-biased in the original dataset. For completely debiased word embeddings, this percentage should be around 50% (with half of the words being (slightly) biased to one gender and the other half of the words to the other). This measure is applied to a set of biased professions provided by Bolukbasi et al. (2016), which originally has a bias score of 85.2%. INLP reduces this score to 73.4% and MP to 74.5%. Both methods thus yield comparable results that reveal that bias is reduced but not completely removed.

TPR-GAP: “gender in the wild”. De-Arteaga et al. (2019) propose an approach that considers the impact of bias on downstream classification tasks. The approach measures to what extent debiasing methods can remove gender bias from a system that predicts a person’s occupation based on a biography describing them. A fair system should have equal performance for members of different classes (Hardt et al., 2016). Specifically, it should not amplify a bias that is present in the label distribution. In the case of gender, the system should perform equally well predicting stereotypically gendered occupations such as surgeon, caretaker, secretary or marine, regardless of whether the biographical texts are about men or women. De-Arteaga et al. (2019) use the GAP^{TPR} score to measure to what degree the performance of a classification system is impacted by bias. The score is calculated as follows: $TPR_{g,y}$ (Equation 1) is the true positive rates for a given profession (y) and gender (g). $TPR-$

GAP $\text{Gap}_{g,y}^{\text{TPR}}$ is the difference (gap) between true positive rates (TPR) of gender g and its opposite gender $\sim g$ in a given occupation y (Equation 2). In order to calculate a single measure for bias in all occupations Romanov et al. (2019) take the root-mean square of the TPR-GAPs for all occupations (Equation 3).

$$\text{TPR}_{g,y} = P[\hat{Y} = y | G = g, Y = y] \quad (1)$$

$$\text{Gap}_{g,y}^{\text{TPR}} = \text{TPR}_{g,y} - \text{TPR}_{\sim g,y} \quad (2)$$

$$\text{GAP}_g^{\text{RMS}} = \sqrt{\frac{1}{|C|} \sum_{y \in C} (\text{GAP}_{g,y}^{\text{TPR}})^2} \quad (3)$$

We obtain 393,423 biographies from Ravfogel et al. (2020) which is a subset of the original corpus of De-Arteaga et al. (2019) describing people with 28 different occupations.⁹ We split the data in the same 65% training, 10% development and 25% test splits as used by De-Arteaga et al. (2019) and Ravfogel et al. (2020). Like Ravfogel et al. (2020), we use logistic classifiers that take one of three representations of the biographies as input: (1) one-hot BOW, (2) averaged FastText embeddings (Joulin et al., 2017) and (3) the last hidden state of BERT over the $[CLS]$ token.

In this setup, debiasing projections are applied as follows: For INLP, each of the three representations is debiased by training a linear classifier on all 28 occupation classes and using the learned vector to calculate the nullspace. Logistic regression in 100 iterations is used for the BOW representations, 150 linear SVM iterations for FastText representations and 300 linear SVM iterations for BERT. We use Scikit Learn (Pedregosa et al., 2011) for all classification approaches. For the MP setup, we create a mean projection for every occupation o (28 in total) by identifying the projection vector P_o based on the mean of all female and the mean of all male biographies with occupation o .

We report the accuracy of predicting the correct profession (Acc.), the root-mean-square of the GAP^{TPR} scores of all professions (GAP^{RMS}) and the correlation between percentage of women in a given profession and the respective GAP^{TPR} scores (Corr.). If debiasing is successful, the classification accuracy should remain high (compared

to before debiasing), while the GAP score should decrease. We measure the correlation between underpredicting women (the GAP^{TPR} score) and the true percentage of women in the set of professions. A lower correlation is preferable. The intuition behind measuring the correlations in this way is to measure the degree to which bias is amplified compared to the distribution of men and women in a profession. For example, a difference of 5% can have different implications depending on the original distribution: It is worse if the predicted percentage of females in a profession goes down to 15% when the true percentage is 20% compared to a true percentage of 48% going to 43%.

The results presented in Table 5 show that overall results of INLP and MP are comparable. MP retains higher accuracy compared to INLP. MP reduces the GAP^{RMS} much more for one-hot BOW, slightly more for FastText embeddings and slightly less for BERT. INLP reduces the correlation a bit more for both FastText and BERT.

Original			
Representation	Acc.	$\text{GAP}_{\text{female}}^{\text{RMS}}$	Corr.
one-hot BOW	77.6%	0.182	0.895
FastText	77.9%	0.172	0.891
BERT	74.7%	0.153	0.852
INLP			
Representation	Acc.	$\text{GAP}_{\text{female}}^{\text{RMS}}$	Corr.
one-hot BOW	77.1	0.111	0.65
FastText	73.6%	0.103	0.493
BERT	74.7%	0.065	0.353
MP			
Representation	Acc.	$\text{GAP}_{\text{female}}^{\text{RMS}}$	Corr.
one-hot BOW	76.67	0.001	0.58
FastText	75.6%	0.092	0.522
BERT	75.2%	0.087	0.395

Table 5: We report on accuracy of the main task which is the occupation classification (Acc.), the TPR-GAP bias measure ($\text{GAP}_{\text{female}}^{\text{RMS}}$) and the correlation (Corr.) of TPR_g and percentage of women for each occupation. You can see the report per vector representation after INLP and MP debiasing method as well as the original representations without debiasing.

⁹De-Arteaga et al. (2019) use 399,000 biographies. The reduction is due to biographies no longer being available when scraping the data.

C Additional Results of Diving Deeper

This appendix provides additional results on the WEAT scores. Figure 9 presents the result of applying random projection iterations after 8 INLP iterations. Figure 10 illustrates when random projections are applied to the original data (without first applying projections for removing gender). These figures illustrate that random projections after 8 INLP iterations mostly lead to a decrease in WEAT score, whereas applying random projection directly to fully biased data increases the scores in the far majority of the cases.

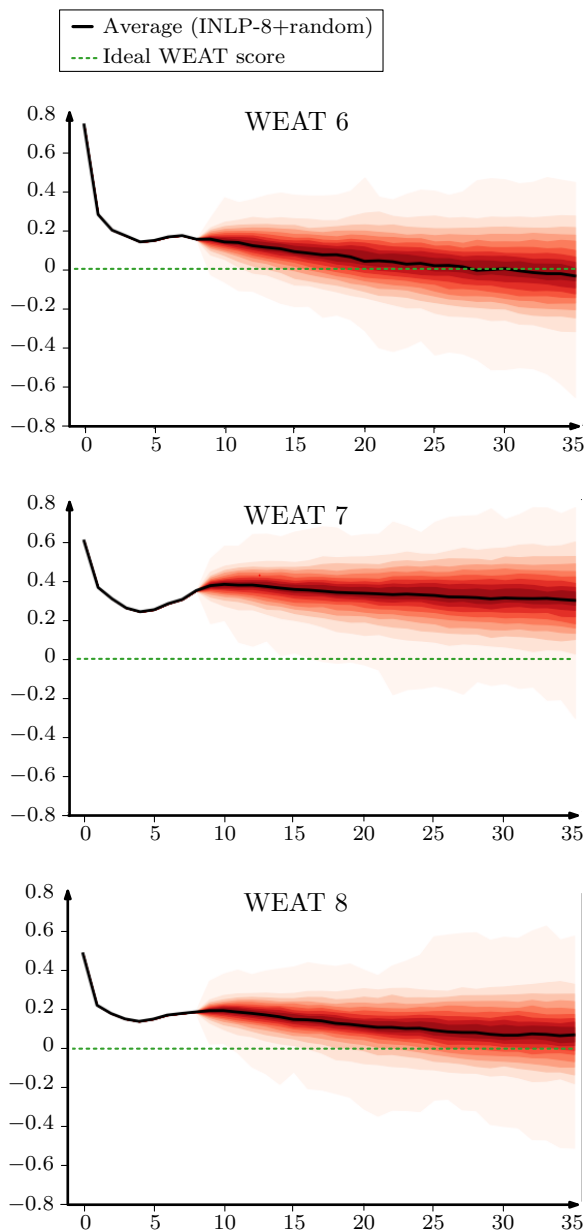


Figure 9: Plot of 500 runs where we first apply INLP 8 times and then continue with 27 iterations of random projections. The shades of red indicate various levels of confidence in WEAT scores (y -axis) after a certain number of iterations (x -axis), where darker red shades reflect higher probabilities of the WEAT score value after a certain number of iterations. The black line is the average WEAT score of all runs and the dotted green line is drawn at $y=0$ (the ideal WEAT score).

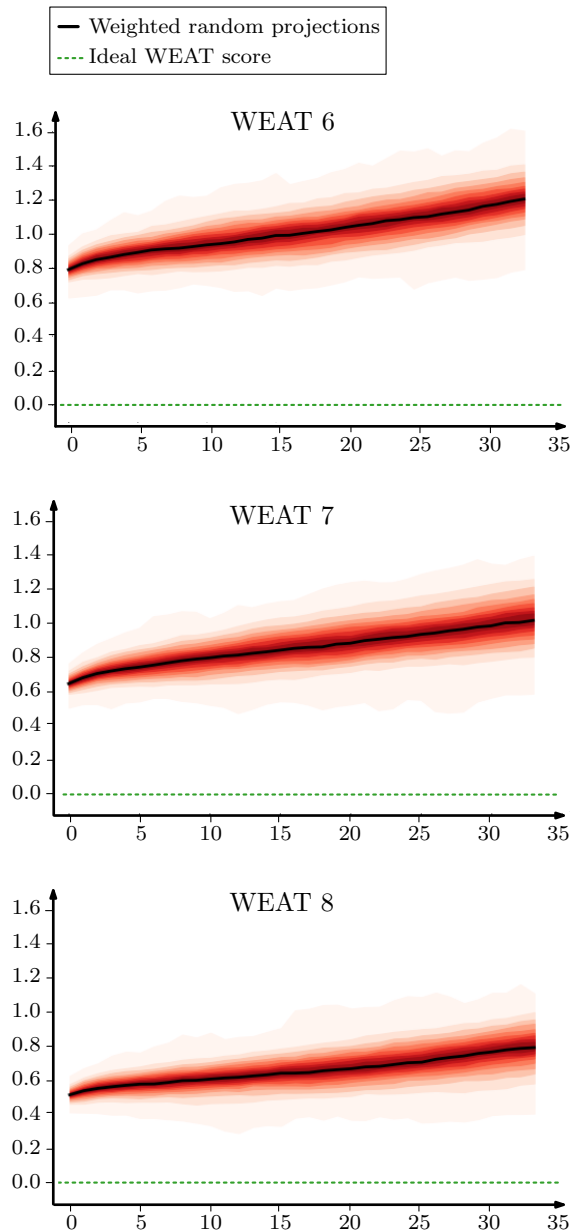


Figure 10: Plot of 500 runs where in every iteration we perform a weighted random projection. The shades of red indicate various levels of confidence in WEAT scores (y -axis) after a certain number of iterations (x -axis), where darker red shades reflect higher probabilities of the WEAT score value after a certain number of iterations. The black line is the average WEAT score of all runs and the dotted green line is drawn at $y=0$ (the ideal WEAT score).

D Changes in the Embedding Space of GloVe

Word	Original neighbors	Neighbors after MP	Neighbors after INLP
order	orders, ordering, purchase	orders, ordering, purchase	orders, ordering, ordered
crack	keygen, cracks, torrent	keygen, cracks, torrent	keygen, cracks, warez
craigslist	ebay, craigslist, ads	ebay, craigslist, ads	ebay, craigslist, freecycle
populations	population, species, communities	population, species, communities	population, species, habitats
epub	ebook, mobi, pdf	ebook, mobi, pdf	mobi, ebook, kindle
finals	semifinals, playoffs, championship	semifinals, playoffs, semifinal	semifinals, semifinal, quarterfinals
installed	install, installing, installation	install, installing, installation	install, installing, installs
identifiable	disclose, identify, identifying	disclose, identify, identifying	disclose, pii, distinguishable
photographs	photograph, photos, images	photograph, photos, images	photograph, images, photos
ta	si, tu, ti	si, tu, na	que, bien, ele
couch	sofa, sitting, bed	sofa, sitting, bed	sofa, couches, loveseat
cooler	coolers, cooling, warmer	coolers, cooling, warmer	coolers, cooling, warmer
becky	debbie, kathy, julie	debbie, kathy, karen	debbie, steph, jen
appreciated	appreciate, greatly, thanks	appreciate, greatly, thanks	appreciate, muchly , thanks
negotiation	negotiating, negotiations, mediation	negotiating, negotiations, mediation	negotiating, negotiations, mediation
initial	subsequent, prior, following	subsequent, prior, following	intial, inital, subsequent
chloe	chanel, emma, lauren	chanel, handbags, handbag	chloe, chanel, handbags
filipino	pinoy, filipinos, philippine	pinoy, filipinos, philippine	filipinos, pinoy, tagalog
relying	rely, relied, relies	rely, relied, relies	rely, relied, relies
perpetual	eternal, continual, irrevocable	eternal, continual, irrevocable	irrevocable, datejust, perpetuity
himself	him, herself, his	herself, him, myself	herself, oneself, he
seaside	beach, beachside, picturesque	beach, beachside, picturesque	beachside, idyllic, seafront
measure	measures, measuring, measured	measures, measuring, measured	measures, measuring, measured
yorkshire	staffordshire, leeds, lancashire	staffordshire, leeds, lancashire	staffordshire, dales , lancashire
merchandise	goods, items, apparel	goods, items, apparel	goods, items, merchandize
sub	subs, k, def	subs, def, k	subs, subbed, sv
tones	tone, hues, muted	tone, hues, muted	tone, polyphonic , muted
therapist	therapists, psychologist, therapy	therapists, therapy, psychologist	therapists, physiotherapist , psychologist
leaned	sighed, smiled, glanced	sighed, leant , smiled	leant, leaning , sighed
tho	nnd, cuz, tlie	nnd, cuz, tlie	nnd, tlio , tlie
lawyers	attorneys, lawyer, attorney	attorneys, lawyer, attorney	attorneys, lawyer, attorney
compile	compiling, compiler, compiles	compiling, compiles, compiler	compiling, compiler, compiles
chord	chords, progressions, guitar	chords, progressions, melody	chords, progressions, voicings
aims	aim, aimed, aiming	aim, aimed, aiming	aim, aimed, aiming
ensure	ensuring, assure, ensures	ensuring, assure, ensures	ensuring, ensures, assure
aerospace	aviation, engineering, automotive	aviation, engineering, automotive	aeronautics , aviation, aeronautical
clubhouse	pool, playground, amenities	pool, playground, amenities	clubhouses , pool, playground
locking	lock, locks, latch	lock, locks, latch	lock, locks, latch
reign	reigns, emperor, throne	reigns, throne, reigned	reigns, reigned , emperor
vulnerable	susceptible, fragile, affected	susceptible, fragile, affected	susceptible, vulnerable , fragile

Table 6: The 3-nearest neighbors of 40 randomly chosen words are shown originally, after mean projection (MP) and after INLP debiasing method. Changed tokens are marked in red.

Word	Original neighbors	Neighbors after MP	Neighbors after INLP
ruth	helen, esther, margaret	esther, helen, margaret	etting, esther, gehrig
charlotte	raleigh, nc, atlanta	raleigh, nc, atlanta	raleigh, greensboro, nc
abigail	hannah, lydia, eliza	hannah, lydia, josiah	hannah, phebe, josiah
sophie	julia, marie, lucy	claire, julia, madeleine	moone, bextor, marceau
nichole	nicole, kimberly, kayla	nicole, kimberly, mya	nicole, mya, heiress
emma	emily, lucy, sarah	emily, watson, sarah	grint, frain, watson
olivia	emma, rachel, kate	wilde, munn, oliver	munn, thirlby, wilde
ava	devine, zoe, isabella	devine, isabella, appellation	viticultural, devine, appellation
isabella	sophia, josephine, isabel	isabel, josephine, henry	rossellini, beeton, ferdinand
sophia	anna, lydia, julia	hagia, antipolis, sofia	hagia, antipolis, topkapi
mia	bella, mamma, mama	bella, mamma, che	bangg, mamma, culpa
amelia	earhart, louisa, caroline	earhart, fernandina, louisa	earhart, fernandina, bedelia
james	john, william, thomas	john, william, thomas	jassie, nightfire, perse
john	james, william, paul	james, william, mary	deere, scatman, betjeman
robert	richard, william, james	richard, william, james	pattinson, mccammon, blacksportsonline
michael	david, mike, brian	david, mike, jackson	micheal, franti, moorcock
william	henry, edward, james	henry, edward, charles	edward, henry, sir
david	stephen, richard, michael	richard, alan, stephen	bisbal, magen, sylvian
richard	robert, william, david	robert, william, john	clayderman, brautigan, rorty
joseph	francis, charles, thomas	mary, francis, charles	joesph, dreamcoat, abboud
thomas	james, william, john	william, james, john	szasz, deshaun, tomy
ariel	sharon, alexis, hanna	israel, israeli, gaza	peterpan, mermaid, cinderella
mike	brian, chris, dave	dave, jim, chris	mignola, birbiglia, dave

Table 7: The 3-nearest neighbors of a selection of inherently gendered names are shown originally, after mean projection (MP) and after INLP debiasing method. Changes to same gender, non-gendered or common nouns are marked in red. Changes to names associated with the opposite gender are marked in blue.