CIENCIAS DE LA COMPUTACIÓN
UNIVERSIDAD DE CHILE

Towards Fairer Word Embeddings: Methodologies for Comparing and Optimizing Bias Mitigation Algorithms

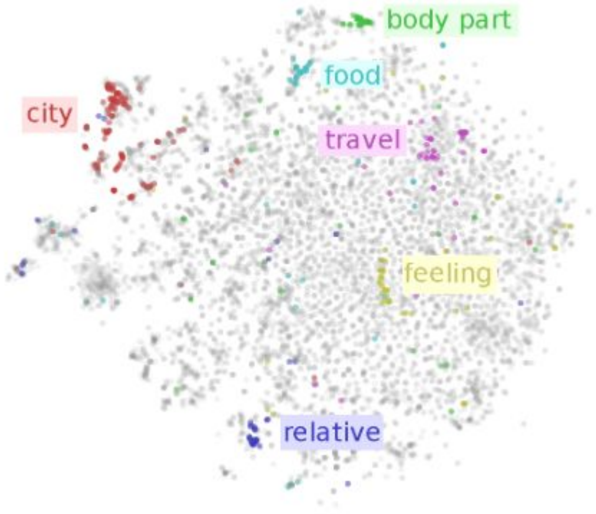
María José Zambrano

mzambran@dcc.uchile.cl

Word Embeddings

➔ Word embedding models are mappings from discrete words to dense continuous vectors.

➔ These models are based on the distributional hypothesis:
  - ◆ Words appearing in similar context have similar meaning.



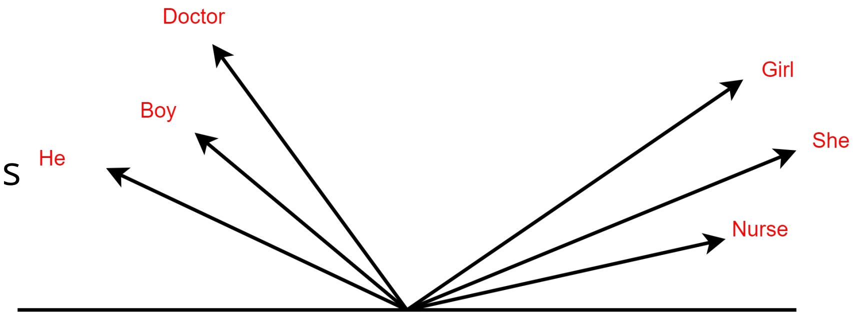
https://colah.github.io/posts/2015-01-Visualizing-Representations/

Bias in Word Embeddings

➔ Word embeddings have been demonstrated to reflect biases inherent in the corpora from which they are trained.

➔ These biases include gender, racial, religious among other.

➔ Leading to unfair representations.

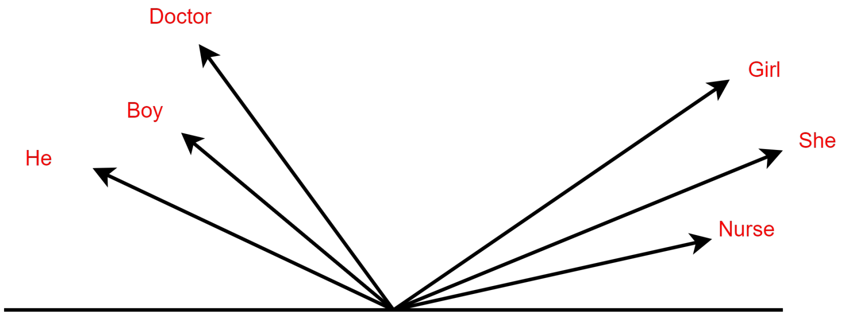


Bias in Word Embeddings

To address this issue, two types of solutions have emerged:

➔ Metrics for quantifying bias levels.

➔ Mitigation algorithms aimed at reducing bias within the model



Bias Measurement

➔ Previous research in the field has introduced various bias measurement metrics for word embedding models.

➔ These metrics share a common goal of quantifying the bias contained in these models but employ distinct methodologies to achieve this objective.

➔ In general, they measure the association between words that define a bias group and words typically associated with that group.

Bias Measurement

➔ Examples of these metrics are Word Embedding Association Test (WEAT), WEAT Effect Size (WEAT ES), Relative Norm Distance (RND).

➔ These metrics, and some others, are standardized and unified within a common interface, facilitating their interchangeability within the WEFE library .

Bias Mitigation

- ➔ Various bias mitigation algorithms have been developed.

- ➔ These algorithms aim to diminish the bias present in word embedding models through diverse approaches.

- ➔ In general, they focus on learning bias from words that define the social groups and adjust the embedding space to ensure that biased words are all at a similar distance from the bias space.

Bias Mitigation

- Examples of these algorithms are: Hard Debias (HD), Double Hard Debias (DHD), Repulsion Attraction Neutralization (RAN) and Half Sibling Regression (HSR).

- The algorithms described above are implemented within a common interface in the WEFE library.

- Previous work has not systematically compared these methods, and there are significant discrepancies and interdependencies between methods and metrics that can affect the reliability of the results.

Comparing Algorithms

Problems when Comparing Algorithms

1. Inconsistencies in normalization transformations.

2. Reliance on different word sets when applying bias mitigation algorithms.

3. Leakage between training words employed by mitigation methods and evaluation words used by metrics.

Vector Normalization

➔ Hard Debias (HD) and Repulsion Attraction Neutralization (RAN), use vector normalization as a preprocessing step in their mitigation process.

➔ We have noticed that just by normalizing the vectors in a model some metrics are affected.

Metrics \ Models	Glove	Glove Normalized
WEAT	0.8446	0.8446
WEAT ES	0.6556	0.6556
RND	0.1832	0.0252
RNSB	0.0859	0.0177
RIPA	0.2274	0.0344
ECT	0.8234	0.8190

➔ This means that comparison between algorithms that normalize vectors and those that do not it is not fair.

Reliance on different word sets

➔ The original implementations of the algorithms demonstrate variability in the selection of words within sets. We contend that this variability introduces additional uncertainty into the observed bias changes.

➔ Particularly relevant when there is a distinction in the words to which the algorithms are applied.

Leakage Between Word Sets

- ➔ We noticed that word sets used by algorithms and metrics overlap.
- ➔ We argue that the overlap between sets may hinder accurate bias measurement and comparison of bias mitigation algorithms.
- ➔ Words used for learning bias mitigation should be excluded from the evaluation to ensure generalization in the measurement, similar to the separation of training and test sets in standard supervised machine learning problems.

Word Interaction

➔ **Bias definition:** refers to a set of word pairs derived from two contrasting identity groups utilized by mitigation algorithms to learn and address the intended bias direction. These words consistently represent male and female groups in bias definition methods (e.g., man-woman, he-she, girl-boy).

➔ **Target:** words are used to denote specific social identity groups defined by criteria such as gender, religion, or race

➔ **Attributes:** include words that represent attitudes, traits, characteristics, occupations, among others. In a fair setting, these attributes should have equal associations with individuals from each social group (e.g., occupations, affective words)

Word Interaction

- ➔ **Gender specific:** Includes words that are associated with gender by definition but do not necessarily define the identity group (e.g., beard, womb,testosterone). These words inherently contain gender-related connotations, so the bias mitigation process is not applied to them. Note that bias definition words are also included in this set.

- ➔ **Objective:** Is the set of words to which the bias mitigation process is applied, which is usually the complement of the gender-specific set. These words are expected to be unrelated to the target identity groups.

Word Interaction

➔ Size of the intersections between the wordsets:

Algorithms \ Metrics	Attributes (6,894)	Target (40)
Objective (398,559)	6,500	0
Bias Definition (44)	0	40
Gender Specific (1,449)	5	40

➔ An instance illustrating intersections between Gender-Specific and Attributes are “maid,” “heroine,” “mistress,” “womanizer,” and “hellion” are identified

Sources of Variability

➔ Selection of words used for training and evaluating the debiasing models.

➔ Hyperparameters of the debiasing models.

➔ The choice of word sets and the fixed combination of hyper-parameters used in bias mitigation and measurement could potentially favor one algorithm over another, thus undermining the fairness of the comparison.

This Research

Research Problems

- ➔ The lack of a thorough examination of word sets, bias metrics, and external variables has resulted in comparisons that often overlook factors affecting bias that are not attributable to the algorithms themselves, leading to unfair comparisons.

- ➔ There is no clear evidence of which algorithms are superior for reducing bias in word embeddings. Until these factors are controlled, the true differences in algorithm performance will remain uncertain.

- ➔ This states the need for a controlled comparison methodology.

Objectives

➔ The general objective of this research is to generate a fair comparison of bias mitigation algorithms to assess real differences in their performance.

➔ To achieve the objective we propose two research lines:

- Straightforward Methodology: Enforce identical word sets, manage method-metric overlap, and apply consistent vector normalization for robust comparability.

- Robust Comparison Framework: Incorporate nested cross-validation, hyper-parameter optimization, and statistical testing, inspired by techniques from supervised learning models to address all sources of variability.

Straightforward Methodology
Comparison Methodology

Vector Normalization

To address the impact of vector normalization on bias measurement, we propose two approaches:

1. Normalize the model before bias mitigation for algorithms that do not perform it
2. reverse the normalization performed by any algorithm (e.g.,HD, RAN) after mitigation by rescaling the resulting vectors to their original norm.

We consider the second approach to be more appropriate as it preserves the information carried by vector length, which is known to contain valuable information within word embeddings.

Standardize Word Sets

- ➔ Considering that the use of different sets introduces variability in the results that is not attributable to the algorithms, we propose standardizing the word sets and using the exact same sets when comparing algorithms.

- ➔ This means that all of the word sets used in algorithms and metrics are composed of exactly the same words when comparing algorithms.

Manage Overlap Between Word Sets

➔ To address this problem concerning the overlap of word sets, we propose implementing constraints on the intersection of the different word sets.

Algorithms \ Metrics	Attributes	Target
Objective	$\|Attributes\|$	$\emptyset$
Bias Definition	$\emptyset$	$\emptyset$
Gender Specific	$\emptyset$	$\|Target\|$

Manage Overlap Between Word Sets

- **Objective/Attributes:** The attributes set should be entirely contained within the objective set to ensure that words expected to be unrelated to social identity groups (i.e., attribute words) are mitigated.

- **Objective/Target:** The target set should not overlap with the objective set. This is crucial because the target words inherently represent specific social identity groups, and applying mitigation techniques to them would directly impact their ability to represent those groups accurately.

- **Bias Definition/Attributes:** These sets are defined as opposites and hence, should not overlap. The bias definition set contains words that define social identity groups (e.g., male and female words), while attribute words are expected to be independent of these criteria.

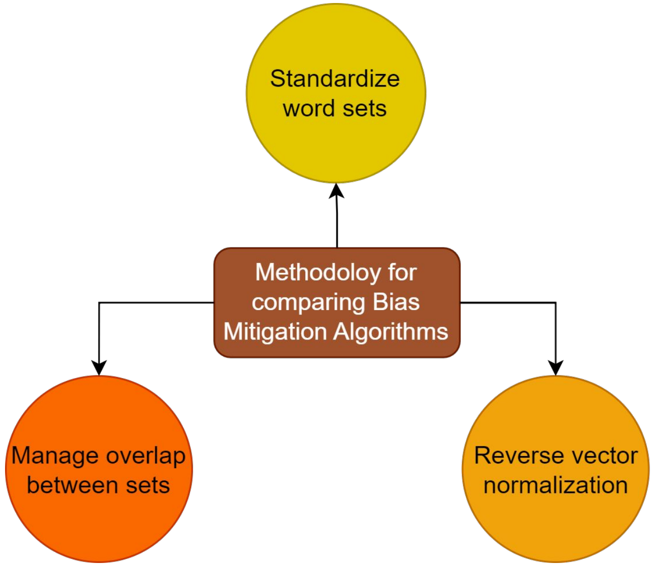
Manage Overlap Between Word Sets

➔ **Bias Definition/Target:** Although both sets contain words that define social identity groups, avoiding overlap between them is important. We expect that mitigation algorithms should generalize beyond the words used to learn the transformation. Assessing bias on the same words used for learning would lead to overly optimistic results. This restriction is analogous to the standard practice of separating training and test data in supervised machine learning.

➔ **Gender Specific/Target:** The target set should be entirely contained within the gender-specific set to avoid bias mitigation on words that define social identity groups.

➔ **Gender Specific/Attributes:** To maintain independence between gender and attributes, the attribute and gender-specific sets should not overlap. This ensures that attribute words, which are intended to be gender-neutral, can be accurately evaluated by the metric after mitigation. This constraint does not affect the generalization of the measurement as mitigation algorithms do not rely on the attribute set for learning the transformation.

Proposed Methodology



Experiments

Experimental Setting

- ➔ For all our experiments, we utilize the glove-wikigigaword-300 model, which is accessible through Gensim.

- ➔ Our baseline setup consists of applying the four bias mitigation algorithms: HD, DHD, RAN, and HSR. These algorithms are applied to the word embedding model employing the default settings from their original implementations.

- ➔ We assess the model's bias levels both before and after mitigation, according to the metrics WEAT, WEAT ES, RND, RNSB, ECT, and RIPA.

Comparing algorithms

➔ We compare the 4 algorithms employing our methodology, for both reversing normalization and normalizing the model before the debias.

➔ We contrast these results with our baseline comparing the change in bias produced by the algorithms.

➔ Utilizing p-values at a significance level of 0.05, we assess the effectiveness of our methodology in reducing bias metric variability across different debiasing methods compared to the baseline.

Baseline methodology

Models / Δ Metrics	HD	DHD	HSR	RAN	σ
WEAT (↓)	-0.756 (1)	-0.058 (4)	-0.647 (3)	-0.677 (2)	0.320
WEAT ES (↓)	-0.519 (1)	-0.030 (4)	-0.145 (3)	-0.428 (2)	0.230
RND (↓)	-0.177 (1)	-0.010 (3)	-0.007 (4)	-0.176 (2)	0.097
RNSB (↓)	-0.094 (1)	-0.027 (3)	0.007 (4)	-0.092 (2)	0.043
RIPA (↓)	-0.221 (1)	-0.014 (4)	-0.197 (3)	-0.213 (2)	0.098
ECT (↑)	0.144 (1)	0.009 (3)	-0.55 (4)	0.132 (2)	0.185
					$\overline{\sigma}$: 0.162

Proposed methodology reversing normalization

Models / Δ Metrics	HD	DHD	HSR	RAN	σ
WEAT (↓)	-0.376 (1)	-0.317 (3)	-0.236 (4)	-0.324 (2)	0.050
WEAT ES (↓)	-0.429 (1)	-0.283 (3)	-0.166 (4)	-0.328 (2)	0.094
RND (↓)	-0.031 (3)	-0.113 (1)	0.027 (4)	-0.038 (2)	0.049
RNSB (↓)	-0.008 (2)	-0.010 (1)	-0.0008 (3)	0.006 (4)	0.006
RIPA (↓)	-0.057 (3)	-0.002 (4)	-0.094 (1)	-0.064 (2)	0.033
ECT (↑)	0.061 (2)	0.027 (3)	-0.152 (4)	0.077 (1)	0.091
					$\overline{\sigma}$: 0.053
					p-value 0.04

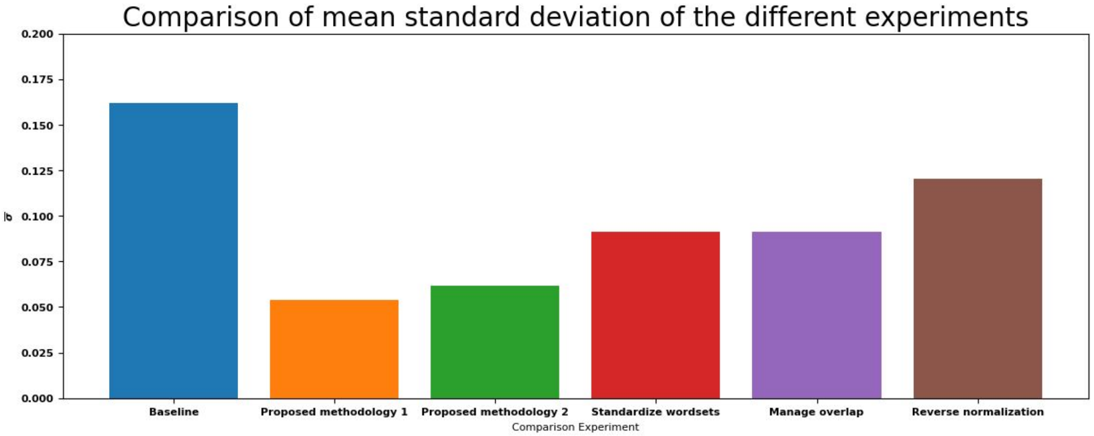
Proposed methodology normalizing the model before debias

Models / Δ Metrics	HD	DHD	HSR	RAN	σ
WEAT (↓)	-0.376 (2)	-0.386 (1)	-0.0125 (4)	-0.324 (3)	0.153
WEAT ES (↓)	-0.429 (1)	-0.426 (2)	-0.005 (4)	-0.328 (3)	0.173
RND (↓)	-0.008 (3)	-0.016 (1)	-0.0004 (4)	-0.010 (2)	0.005
RNSB (↓)	-0.003 (2)	-0.005 (1)	0.0004 (4)	-0.003 (3)	0.001
RIPA (↓)	-0.013 (2)	-0.013 (1)	-0.0009 (4)	-0.012 (3)	0.005
ECT (↑)	0.058 (2)	0.039 (3)	-0.009 (4)	0.078 (1)	0.032
					$\overline{\sigma}$: 0.061
					p-value 0.08

Comparing algorithms

- ➔ Applying our methodology reduces the performance gap between algorithms and enhances DHD's bias reduction performance. Reverting normalization significantly reduces variability in bias reduction among algorithms, ensuring more objective evaluations.

- ➔ Conversely, normalizing the model before normalization implementation results in a smaller and statistically insignificant reduction in standard deviation. Our methodology ensures fair comparison of algorithms, with results indicating similar bias reduction among algorithms.



Analysis of Isolated Components

➔ We perform an isolated component analysis of our proposed methodology.

➔ The aim is to methodically evaluate the impact of each component of the methodology.

Standardize word sets					
Models / Δ Metrics	HD	DHD	HSR	RAN	σ
WEAT (↓)	-0.6731 (2)	-0.6484 (3)	-0.385 (4)	-0.6774 (1)	0.122
WEAT ES (↓)	-0.4086 (2)	-0.3202 (3)	-0.1321 (4)	-0.4289 (1)	0.117
RND (↓)	-0.177 (1)	-0.0609 (3)	-0.0365 (4)	-0.1761 (2)	0.064
RNSB (↓)	-0.0705 (2)	-0.0371 (3)	0.0464 (4)	-0.0755 (1)	0.048
RIPA (↓)	-0.2154 (1)	-0.1457 (3)	-0.1046 (4)	-0.2133 (2)	0.046
ECT (↑)	0.1431 (1)	0.1171 (3)	-0.2201 (4)	0.1326 (2)	0.152
					$\overline{\sigma}$: 0.092
					p-value 0.16
Manage overlap between sets					
Models / Δ Metrics	HD	DHD	HSR	RAN	σ
WEAT (↓)	-0.4580 (1)	-0.0290 (4)	-0.2362 (3)	-0.3245 (2)	0.155
WEAT ES (↓)	-0.5555 (1)	-0.0229 (4)	-0.1660 (3)	-0.3287 (2)	0.198
RND (↓)	-0.3246 (1)	-0.0058 (3)	0.0278(4)	-0.3114 (2)	0.164
RNSB (↓)	-0.0537 (1)	0.0126 (4)	0.0107 (3)	-0.0528 (2)	0.032
RIPA (↓)	-0.1986 (1)	-0.0130 (4)	-0.0946 (3)	-0.1902 (2)	0.076
ECT (↑)	0.0850 (1)	0.0036 (3)	-0.1526 (4)	0.0801 (2)	0.096
					$\overline{\sigma}$: 0.120
					p-value 0.23
Reverse vector normalization					
Models / Δ Metrics	HD	DHD	HSR	RAN	σ
WEAT (↓)	-0.7561 (1)	-0.0587 (4)	-0.6479 (3)	-0.6774 (2)	0.277
WEAT ES (↓)	-0.5193 (1)	-0.0301 (4)	-0.1456 (3)	-0.4289 (2)	0.344
RND (↓)	-0.1527 (1)	-0.0100 (3)	-0.0076 (4)	-0.0956 (2)	0.061
RNSB (↓)	-0.0142 (2)	-0.0006 (3)	0.0240 (4)	-0.0156 (1)	0.015
RIPA (↓)	-0.1902 (2)	-0.0148 (4)	-0.1971 (1)	-0.1265 (3)	0.073
ECT (↑)	0.1458 (1)	0.0090 (3)	-0.2552(4)	0.1340 (2)	0.161
					$\overline{\sigma}$: 0.155
					p-value 0.92

Analysis of Isolated Components

➔ An important observation is the improvement in bias mitigation by DHD in the standardized setting, consistent with our methodology's results. This affirms that standardizing word sets significantly contributes to this improvement.

➔ Controlling word sets enhances the comparability of bias mitigation algorithms. Our focus is solely on managing overlap between algorithms and metrics without altering word sets otherwise.

Standardize word sets					
Models / Δ Metrics	HD	DHD	HSR	RAN	σ
WEAT (↓)	-0.6731 (2)	-0.6484 (3)	-0.385 (4)	-0.6774 (1)	0.122
WEAT ES (↓)	-0.4086 (2)	-0.3202 (3)	-0.1321 (4)	-0.4289 (1)	0.117
RND (↓)	-0.177 (1)	-0.0609 (3)	-0.0365 (4)	-0.1761 (2)	0.064
RNSB (↓)	-0.0705 (2)	-0.0371 (3)	0.0464 (4)	-0.0755 (1)	0.048
RIPA (↓)	-0.2154 (1)	-0.1457 (3)	-0.1046 (4)	-0.2133 (2)	0.046
ECT (↑)	0.1431 (1)	0.1171 (3)	-0.2201 (4)	0.1326 (2)	0.152
					$\overline{\sigma}$: 0.092
					p-value 0.16

Manage overlap between sets					
Models / Δ Metrics	HD	DHD	HSR	RAN	σ
WEAT (↓)	-0.4580 (1)	-0.0290 (4)	-0.2362 (3)	-0.3245 (2)	0.155
WEAT ES (↓)	-0.5555 (1)	-0.0229 (4)	-0.1660 (3)	-0.3287 (2)	0.198
RND (↓)	-0.3246 (1)	-0.0058 (3)	0.0278(4)	-0.3114 (2)	0.164
RNSB (↓)	-0.0537 (1)	0.0126 (4)	0.0107 (3)	-0.0528 (2)	0.032
RIPA (↓)	-0.1986 (1)	-0.0130 (4)	-0.0946 (3)	-0.1902 (2)	0.076
ECT (↑)	0.0850 (1)	0.0036 (3)	-0.1526 (4)	0.0801 (2)	0.096
					$\overline{\sigma}$: 0.120
					p-value 0.23

Reverse vector normalization					
Models / Δ Metrics	HD	DHD	HSR	RAN	σ
WEAT (↓)	-0.7561 (1)	-0.0587 (4)	-0.6479 (3)	-0.6774 (2)	0.277
WEAT ES (↓)	-0.5193 (1)	-0.0301 (4)	-0.1456 (3)	-0.4289 (2)	0.344
RND (↓)	-0.1527 (1)	-0.0100 (3)	-0.0076 (4)	-0.0956 (2)	0.061
RNSB (↓)	-0.0142 (2)	-0.0006 (3)	0.0240 (4)	-0.0156 (1)	0.015
RIPA (↓)	-0.1902 (2)	-0.0148 (4)	-0.1971 (1)	-0.1265 (3)	0.073
ECT (↑)	0.1458 (1)	0.0090 (3)	-0.2552(4)	0.1340 (2)	0.161
					$\overline{\sigma}$: 0.155
					p-value 0.92

Analysis of Isolated Components

- ➔ Although controlling word set overlap lacks statistical significance, its primary goal extends beyond comparability. It aims to ensure accurate bias reduction measurement by eliminating dependencies between methods and metrics.

- ➔ While examining consistent vector normalization transformations, this specific setting does not significantly impact the results. Nonetheless, vector normalization remains crucial for fair algorithm comparison, as it ensures that differences in bias are not influenced by normalization but are solely attributed to algorithm application.

Standardize word sets

Models / Δ Metrics	HD	DHD	HSR	RAN	σ
WEAT (↓)	-0.6731 (2)	-0.6484 (3)	-0.385 (4)	-0.6774 (1)	0.122
WEAT ES (↓)	-0.4086 (2)	-0.3202 (3)	-0.1321 (4)	-0.4289 (1)	0.117
RND (↓)	-0.177 (1)	-0.0609 (3)	-0.0365 (4)	-0.1761 (2)	0.064
RNSB (↓)	-0.0705 (2)	-0.0371 (3)	0.0464 (4)	-0.0755 (1)	0.048
RIPA (↓)	-0.2154 (1)	-0.1457 (3)	-0.1046 (4)	-0.2133 (2)	0.046
ECT (↑)	0.1431 (1)	0.1171 (3)	-0.2201 (4)	0.1326 (2)	0.152
					$\overline{\sigma}$: 0.092
					p-value 0.16

Manage overlap between sets

Models / Δ Metrics	HD	DHD	HSR	RAN	σ
WEAT (↓)	-0.4580 (1)	-0.0290 (4)	-0.2362 (3)	-0.3245 (2)	0.155
WEAT ES (↓)	-0.5555 (1)	-0.0229 (4)	-0.1660 (3)	-0.3287 (2)	0.198
RND (↓)	-0.3246 (1)	-0.0058 (3)	0.0278(4)	-0.3114 (2)	0.164
RNSB (↓)	-0.0537 (1)	0.0126 (4)	0.0107 (3)	-0.0528 (2)	0.032
RIPA (↓)	-0.1986 (1)	-0.0130 (4)	-0.0946 (3)	-0.1902 (2)	0.076
ECT (↑)	0.0850 (1)	0.0036 (3)	-0.1526 (4)	0.0801 (2)	0.096
					$\overline{\sigma}$: 0.120
					p-value 0.23

Reverse vector normalization

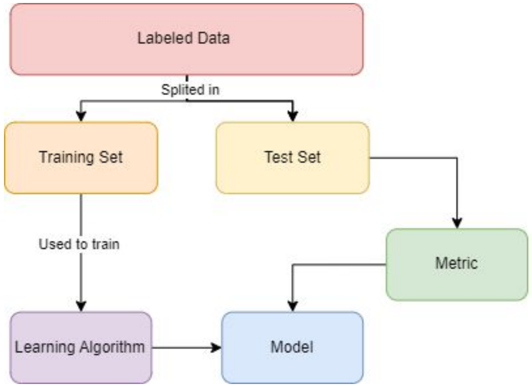
Models / Δ Metrics	HD	DHD	HSR	RAN	σ
WEAT (↓)	-0.7561 (1)	-0.0587 (4)	-0.6479 (3)	-0.6774 (2)	0.277
WEAT ES (↓)	-0.5193 (1)	-0.0301 (4)	-0.1456 (3)	-0.4289 (2)	0.344
RND (↓)	-0.1527 (1)	-0.0100 (3)	-0.0076 (4)	-0.0956 (2)	0.061
RNSB (↓)	-0.0142 (2)	-0.0006 (3)	0.0240 (4)	-0.0156 (1)	0.015
RIPA (↓)	-0.1902 (2)	-0.0148 (4)	-0.1971 (1)	-0.1265 (3)	0.073
ECT (↑)	0.1458 (1)	0.0090 (3)	-0.2552(4)	0.1340 (2)	0.161
					$\overline{\sigma}$: 0.155
					p-value 0.02

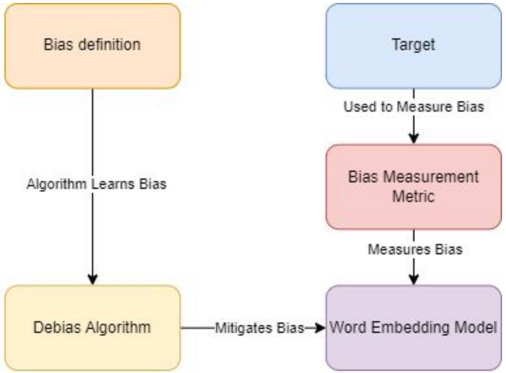
Robust Comparison Framework

Bias Mitigation and Supervised Learning

➔ In supervised learning, the goal is to train a model to predict outputs using a set of labeled data. The process involves dividing this data into a training set for model training and a test set for evaluation, with performance measured through metrics such as accuracy and precision.

➔ In contrast, bias mitigation focuses on reducing bias in word embeddings. This involves using sets of word pairs to define and measure bias. The process similarly involves training on a set (bias definition) and evaluating on a target set, with performance assessed using bias measurement metrics like WEAT and RND.





Hyper-Parameters

➔ Bias mitigation algorithms, much like supervised learning models, have hyper-parameters that influence their performance. These hyper-parameters play a crucial role in determining the effectiveness of bias reduction.

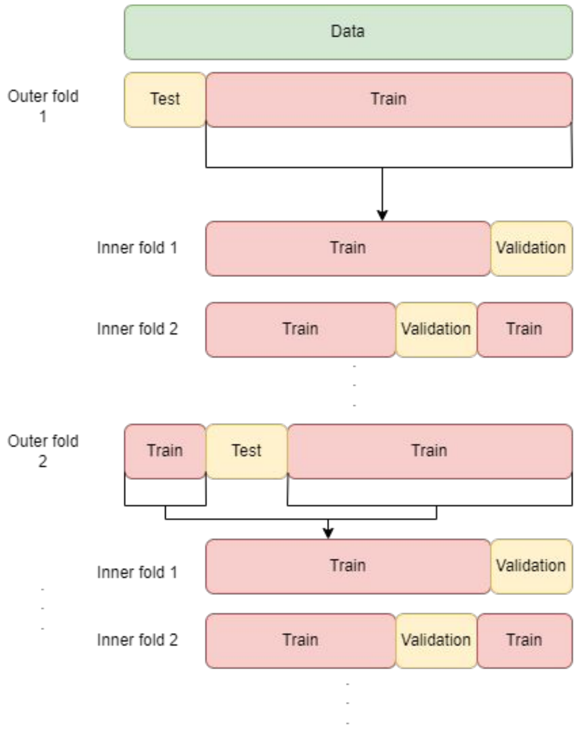
Algorithm	Hyper-parameters
HD	$svd\_solvers$
DHD	$svd\_solvers$
HSR	Alpha
RAN	Epochs, Theta, Neighbours, Learning Rate, Weights

Nested Cross Validation

➔ Nested cross-validation optimizes hyper-parameters by evaluating all combinations using a grid search.

➔ Data is divided into outer folds (training and test sets) and, within each fold, further split into inner folds for detailed hyper-parameter tuning.

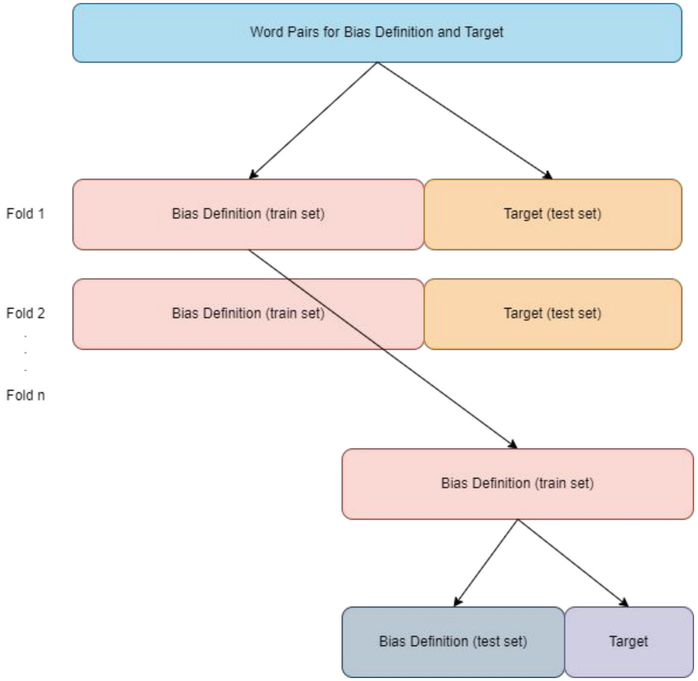
➔ Models are trained and tested on these inner folds to find the best hyper-parameters, which are then applied in the outer fold. This method prevents overfitting and ensures generalization to new dat.



Adapting Nested Cross Validation

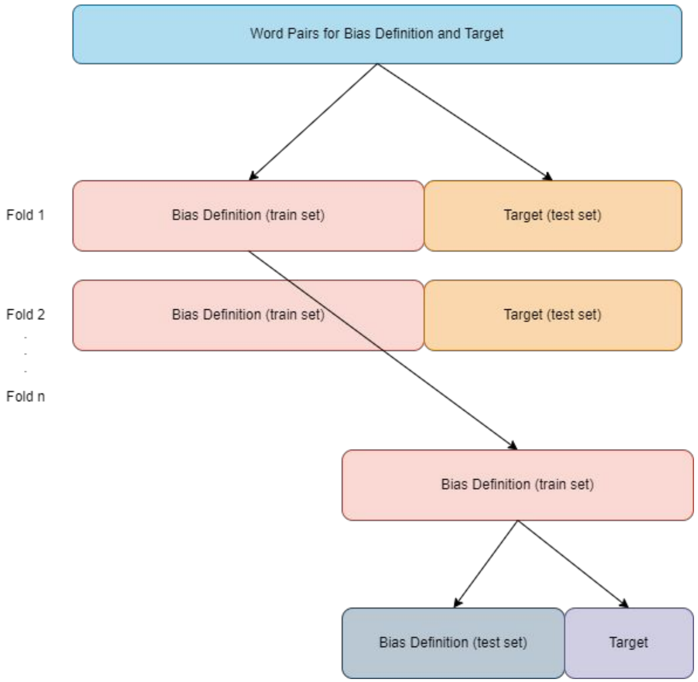
➔ Adapt Nested Cross-Validation: Utilize nested cross-validation, a technique borrowed from supervised learning, to optimize hyper-parameters for bias mitigation algorithms.

➔ This method ensures robust comparison and helps in selecting optimal hyper-parameters for maximum bias reduction.



Adapting Nested Cross Validation

- ➔ Use a set of word pairs as the dataset, similar to labeled data in supervised learning.
- ➔ Grid Search: Define and test a range of hyper-parameters using nested cross-validation. This involves dividing the data into 10 folds, further splitting each fold, and iterating over hyper-parameter combinations.
- ➔ Performance Evaluation: For each fold, apply the optimal hyper-parameters to debias the dataset and measure bias reduction.



Comparison Process

- ➔ Analyze results from all folds to ensure robust performance evaluation and apply statistical tests.

- ➔ Use the corrected paired t-test to compare bias reduction between models, addressing the non-independence of cross-validation folds and ensuring accurate, reliable results.

Experiments

Experimental Setting

- ➔ We adhere to the approach outlined in our straightforward comparison methodology.
- ➔ Utilizing the glove-wikigigaword-300 model.
- ➔ Assessing the model’s bias levels both before and after mitigation using the WEAT, WEAT-ES, RND, RNSB, ECT, and RIPA metrics.
- ➔ Using the same word sets, while extending those used for bias definition and target selection.

Hyper-parameter	**Values**
	HD
Solver	auto, full, arpack, randomized
	DHD
Solver	auto, full, arpack, randomized
	HSR
Alpha	0,10,30,60,80
	RAN
Neighbours	50,100,200
Theta	0.002,0.005,0.1

Experimental Setting

- ➔ We conduct experiments utilizing each of the six bias measurement metrics previously described.

- ➔ We define the hyper-parameter grids as shown in in the table.

Hyper-parameter	**Values**
	HD
Solver	auto, full, arpack, randomized
	DHD
Solver	auto, full, arpack, randomized
	HSR
Alpha	0,10,30,60,80
	RAN
Neighbours	50,100,200
Theta	0.002,0.005,0.1

Comparing Algorithms

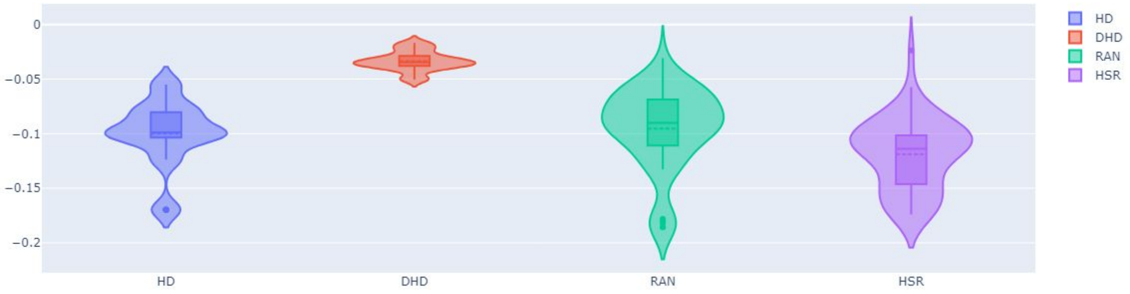
- The table presents mean bias reduction and standard deviations, with significance determined by a corrected paired t-test (significance level 0.05).
- The results indicate that none of the newer algorithms outperform Hard Debias. Double Hard Debias and Half-Sibling Regression generally perform worse, while Repulsion Attraction Neutralization is statistically equivalent to Hard Debias. This suggests that Hard Debias remains as effective as any newer approaches.

Algorithm	Δ Weat	Δ RND	Δ RIPA	Δ RNSB	Δ ECT	Δ WEATES
WEAT						
HD (Baseline)	-0.1 ± 0.031	-0.234 ± 0.102	-0.186 ± 0.035	-0.073 ± 0.024	0.14 ± 0.113	-0.325 ± 0.114
DHD	-0.033 ± 0.009 (↓)	-0.024 ± 0.011 (↓)	-0.035 ± 0.012 (↓)	-0.004 ± 0.012 (↓)	0.008 ± 0.026 (↓)	-0.056 ± 0.036 (↓)
RAN	-0.099 ± 0.037 (=)	-0.233 ± 0.101 (=)	-0.19 ± 0.036 (↑)	-0.074 ± 0.034 (=)	0.15 ± 0.123 (=)	-0.341 ± 0.129 (=)
HSR	-0.124 ± 0.032 (↑)	-0.022 ± 0.092 (↓)	-0.157 ± 0.036 (↓)	-0.042 ± 0.033 (↓)	-0.151 ± 0.082 (↑)	-0.229 ± 0.082 (=)
RNSB						
HD (Baseline)	-0.1 ± 0.031	-0.234 ± 0.102	-0.186 ± 0.035	-0.072 ± 0.026	0.14 ± 0.113	-0.325 ± 0.114
DHD	-0.033 ± 0.009 (↓)	-0.024 ± 0.011 (↓)	-0.035 ± 0.012 (↓)	0.002 ± 0.018 (↓)	0.008 ± 0.025 (↓)	-0.056 ± 0.036 (↓)
RAN	-0.096 ± 0.037 (=)	-0.234 ± 0.102 (=)	-0.19 ± 0.035 (↑)	-0.075 ± 0.033 (=)	0.154 ± 0.108 (=)	-0.351 ± 0.152 (=)
HSR	-0.124 ± 0.031 (↑)	-0.02 ± 0.09 (↓)	-0.156 ± 0.036 (↓)	-0.04 ± 0.031 (↓)	-0.148 ± 0.08 (↓)	-0.248 ± 0.095 (=)
RIPA						
HD (Baseline)	-0.1 ± 0.031	-0.234 ± 0.102	-0.186 ± 0.035	-0.072 ± 0.022	0.14 ± 0.113	-0.325 ± 0.114
DHD	-0.033 ± 0.009 (↓)	-0.024 ± 0.011 (↓)	-0.035 ± 0.012(↓)	-0.003 ± 0.013 (↓)	0.008 ± 0.026 (↓)	-0.056 ± 0.036 (↓)
RAN	-0.096 ± 0.041 (=)	-0.233 ± 0.101 (=)	-0.191 ± 0.034 (↑)	-0.073 ± 0.034 (=)	0.144 ± 0.133 (=)	-0.358 ± 0.158 (=)
HSR	-0.125 ± 0.03 (↑)	-0.02 ± 0.09 (↓)	-0.157 ± 0.036 (↓)	-0.039 ± 0.035 (↓)	-0.143 ± 0.077 (↓)	-0.261 ± 0.072 (=)

Comparing Algorithms

➔ Another key insight from the results is that the choice of optimization metric has minimal impact, as all metrics seem to guide the optimization process in a similar manner. This suggests that the metrics provide comparable information about the bias in embedding models, making them interchangeable in the optimization process.

➔ While Double Hard Debias (DHD) is the least effective at mitigating bias, it yields the most consistent results according to the WEAT metric.



Comparing Algorithms

- ➔ In the table we show the most frequently selected optimal hyper-parameter combinations for each algorithm.

- ➔ For Hard Debias (HD) and Double Hard Debias (DHD), the optimal solver is "auto," which is also the default value. However, for Half-Sibling Regression (HSR) and Repulsion Attraction Neutralization (RAN), the best hyper-parameters differ from the original suggestions by their authors.

- ➔ For HSR, the optimal value for α is 0, effectively reducing the algorithm's regression to a standard linear regression, contrary to the recommended value of 60 by the method's authors

Algorithm	Hyper-parameter	Value
HD	solver	auto
DHD	solver	auto
HSR	alpha	0
RAN	theta,neighbour	0.02,200

Conclusions

Conclusions

- ➔ We addressed the concerns regarding the comparison of bias mitigation algorithms by focusing on key aspects such as word sets and pre-processing steps, including vector normalization. Additionally, we leveraged the similarities between supervised learning and bias mitigation by employing techniques like nested cross-validation to manage variability arising from word sets and hyper-parameters.

- ➔ Our experiments found that controlling variables leads to more consistent algorithm performance, revealing that none of the newer algorithms significantly outperforms the original Hard Debiasing approach by Bolukbasi et al. This aligns with previous findings, showing that controlled comparisons often uncover smaller differences in performance than anticipated.

- ➔ We hope this methodology encourages fairer comparisons in bias mitigation research. Future work will extend this approach to large language models and other languages, promoting fairer representations across different linguistic contexts.

References

[1] Aylin Caliskan, Joanna J Bryson, and Arvind Narayanan. 2017. Semantics derived automatically from language corpora contain human-like biases. Science, 356(6334):183–186.
[2] Nikhil Garg, Londa Schiebinger, Dan Jurafsky, and James Zou. 2018. Word embeddings quantify 100 years of gender and ethnic stereotypes. Proceedings of the National Academy of Sciences, 115(16):E3635–E3644.
[3] Pablo Badilla, Felipe Bravo-Marquez, and Jorge Pérez. 2020. Wefe: The word embeddings fairness evaluation framework. In Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20, pages 430– 436. International Joint Conferences on Artificial Intelligence Organization.
[4] 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.
[5] Tianlu Wang, Xi Victoria Lin, Nazneen Fatema Rajani, Bryan McCann, Vicente Ordonez, and Caiming Xiong. 2020. Double-hard debias: Tailoring word embeddings for gender bias mitigation. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 5443–5453, Online. Association for Computational Linguistics.
[6] Vaibhav Kumar, Tenzin Singhay Bhotia, and Tanmoy Chakraborty. 2020. Nurse is closer to woman than surgeon? mitigating gender-biased proximities in word embeddings. Transactions of the Association for Computational Linguistics, 8:486–503.
[7] Zekun Yang and Juan Feng. 2020. A causal inference method for reducing gender bias in word embedding relations. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 34, pages 9434–9441.
[8] Pablo Badilla, Felipe Bravo-Marquez, and Jorge Pérez. 2020. Wefe: The word embeddings fairness evaluation framework. In Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20, pages 430– 436. International Joint Conferences on Artificial Intelligence Organization.
[9] Kawin Ethayarajh, David Duvenaud, and Graeme Hirst. 2019. Towards understanding linear word analogies. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 3253–3262, Florence, Italy. Association for Computational Linguistics.

CIENCIAS DE LA COMPUTACIÓN
UNIVERSIDAD DE CHILE

Towards Fairer Word Embeddings: Methodologies for Comparing and Optimizing Bias Mitigation Algorithms

María José Zambrano