# VERB: Visualizing and Interpreting Bias Mitigation Techniques Geometrically for Word Representations

ARCHIT RATHORE, University of Utah, USA
SUNIPA DEV, University of California, Los Angeles, USA
JEFF M. PHILLIPS, University of Utah, USA
VIVEK SRIKUMAR, University of Utah, USA
YAN ZHENG, VISA Research, USA
CHIN-CHIA MICHAEL YEH, VISA Research, USA
JUNPENG WANG, VISA Research, USA
WEI ZHANG, VISA Research, USA
BEI WANG, University of Utah, USA

Word vector embeddings have been shown to contain and amplify biases in the data they are extracted from. Consequently, many techniques have been proposed to identify, mitigate, and attenuate these biases in word representations. In this paper, we utilize interactive visualization to increase the interpretability and accessibility of a collection of state-of-the-art debiasing techniques. To aid this, we present the Visualization of Embedding Representations for deBiasing ("VERB") system, an open-source web-based visualization tool that helps users gain a technical understanding and visual intuition of the inner workings of debiasing techniques, with a focus on their geometric properties. In particular, VERB offers easy-to-follow examples that explore the effects of these debiasing techniques on the geometry of high-dimensional word vectors. To help understand how various debiasing techniques change the underlying geometry, VERB decomposes each technique into interpretable sequences of primitive transformations and highlights their effect on the word vectors using dimensionality reduction and interactive visual exploration. VERB is designed to target natural language processing (NLP) practitioners who are designing decision-making systems on top of word embeddings, and also researchers working with the fairness and ethics of machine learning systems in NLP. It can also serve as a visual medium for education, which helps an NLP novice understand and mitigate biases in word embeddings.

CCS Concepts: • **Human-centered computing → Visualization toolkits**.

Additional Key Words and Phrases: Responsible XAI, debiasing, visual data exploration, distributed representations, bias mitigation, model interpretation, ethics

Authors' addresses: Archit Rathore, University of Utah, Salt Lake City, UT, USA, archit@cs.utah.edu; Sunipa Dev, University of California, Los Angeles, Los Angeles, CA, USA, sunipadev@gmail.com; Jeff M. Phillips, University of Utah, Salt Lake City, UT, USA, jeffp@cs.utah.edu; Vivek Srikumar, University of Utah, Salt Lake City, UT, USA, svivek@cs.utah.edu; Yan Zheng, VISA Research, Palo Alto, CA, USA, yazheng@visa.com; Chin-Chia Michael Yeh, VISA Research, Palo Alto, CA, USA, miyeh@visa.com; Junpeng Wang, VISA Research, Palo Alto, CA, USA, junpenwa@visa.com; Wei Zhang, VISA Research, Palo Alto, CA, USA, wzhan@visa.com; Bei Wang, University of Utah, Salt Lake City, UT, USA, beiwang@sci.utah.edu.

**ACM Reference Format:**
Archit Rathore, Sunipa Dev, Jeff M. Phillips, Vivek Srikumar, Yan Zheng, Chin-Chia Michael Yeh, Junpeng Wang, Wei Zhang, and Bei Wang. 2022. VERB: Visualizing and Interpreting Bias Mitigation Techniques Geometrically for Word Representations. *ACM Trans. Interact. Intell. Syst.* 37, 4, Article 111 (August 2022), 34 pages. https://doi.org/XXXXXXX.XXXXXXX

## 1 INTRODUCTION

Complicated and massive data sets are becoming more and more commonly represented as vectorized embeddings. They are part of a de facto model for words in word vector embeddings such as *Word2Vec* [51] and *GloVe* [54], and are becoming commonplace for other data types such as graphs, spatial regions, and financial data. These vectorized embeddings (referred to as *representations*) directly capture the similarity between objects. Additional structures arise implicitly from these representations, such as linear subspaces that capture concepts (e.g., gender, occupation, and nationality) among word vectors. Moreover, these representations permit easy integration into machine learning (ML) tasks.

A downside of these representations is that their high-dimensional nature obscures easy interpretation, at least not without much additional effort. Although these representations do not have explicit agendas, they can nevertheless encode *biases* through data imbalance and other hidden factors. We must emphasize that *biases* are multifaceted. In this paper, *biases* (in the context of word representations) refer to stereotypical associations between words or groups of words that may cause *representational harm* (i.e., the subordination of a certain social group along the lines of identity). A common example is the propensity of male-leaning words associating more strongly with professions seen higher in the social ladder than female-leaning words. This propensity may have negative repercussions for ML systems using these representations. When bias is manifested in the associations of a stereotypical nature that are expressed in word representations, it is often necessary to *modify* these representations to mitigate such a bias [5, 16] to support fair ML. Modification of vectorized representations is also useful in normalization and alignment tasks.

Visualization tools for these high-dimensional word vector representations exist, and are overviewed in Sect. 2. However, many of the previous approaches are *passive* and only allow a user to *inspect*, but not *modify* the representations. Furthermore, they do not provide a visual interface that helps a user *verify* the effects of a bias mitigation mechanism. In contrast, the tool we present – *VERB* (Visualization of Embedding Representations for deBiasing) – is *active*, and it allows a user to modify the embedding while visually exploring it. Once a modification is visually verified, the results can be exported for downstream tasks.

In particular, VERB allows for easy understanding and use of methods to debias word vector embeddings. Specifically, VERB enables users to apply a collection of post-processing debiasing techniques [5, 14, 16, 59] to static non-contextual word embeddings from methods such Word2Vec [51] and GloVe [54], which enables interpretation and comparison of the various methods. For instance, a user can easily observe the difference between the original *Hard Debiasing* approach [5] or the newer and simpler *Linear Projection* method [16] using VERB.

Another often overlooked aspect is the comparison of how concept subspaces are found. Indeed, there are several methods based on PCA, or derived from clustering or classification. In fact, by inspecting the surprising differences between these approaches, we devise a new interactive approach toward subspace identification that outperforms all prior methods. VERB allows users to insert a human-in-the-loop of this process to dynamically improve the result, and verify it is doing what is intended.

Finally, we note that VERB is applicable beyond word vector embeddings. Any vectorized representation can easily be loaded, inspected, actively modified, and outputted. We demonstrate this with a use-case of analyzing merchant

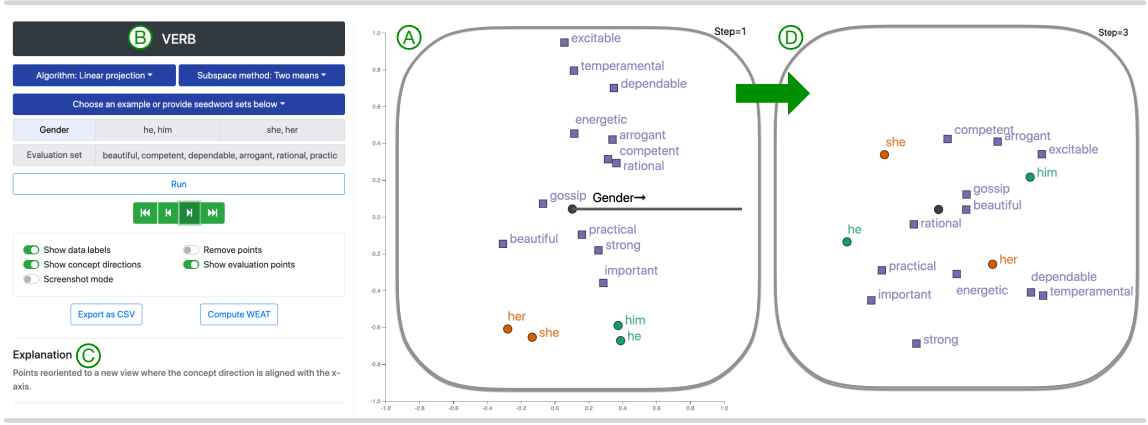


Fig. 1. With VERB, users can explore the high-dimensional word representations interactively before, during, and after applying bias mitigation techniques. (A) **Embedding View** highlights a subset of word embeddings using dimensionality reduction and visualizes the step-by-step transformations of their embeddings across various debiasing techniques. (B) **Control Panel** enables users to configure each debiasing technique and provides controls to iterate through each step of the transformation. (C) **Explanation Panel** gives a step-by-step description of the transformation. In this example, in the Embedding View, word representations are reoriented with the concept direction (Gender) along the x-axis. They are then transformed by a debiasing technique called the Linear Projection that removes this concept. These transformed representations are visualized again in (D) after a new dimensionality reduction, where there is no longer a well-defined gender concept direction, and hence the stereotypical associations are mitigated.

association from a global payment company, which provides new insights that are nebulous or hidden before the use of VERB.

**Contributions.** To summarize our contributions:

- VERB enables NLP and ML researchers to visually analyze the biases in trained static non-contextual word embeddings through a new modular and geometry-preserving approach.
- VERB provides a no-code, accessible pipeline that allows users to apply various post-processing debiasing methods and visually inspect the effect of the debiasing process on the word embeddings.
- VERB has helped spur the development of a new, iterative subspace identification method.
- VERB dually functions as a pedagogical tool that allows educators and fairness/ethics researchers to present biases in a visually interactive form. In particular, it has served as a visual medium for educating attendees at ML and data mining venues (AAAI and KDD), and high school students via an Engineering Summer Camp and a Discover Engineering exhibit.
- VERB is open-source, available at https://github.com/tdavislab/verb.

An overview of the VERB interface is shown in Fig. 1.

**Trigger Warning: This paper contains examples of biases and stereotypes seen in society and in language representations. These examples may be potentially triggering and offensive. The inclusion of these examples is meant to bring light to and mitigate these biases, and it is not an endorsement.**

## 2 RELATED WORK

### 2.1 Visual Analytics for ML Interpretability

**Visual analytics for ML models.** With the recent success of machine learning (ML), a growing number of visual analytics works have been proposed for the interpretation of ML models [3, 10, 36, 41, 55, 61, 77, 80]. Based on the analysis focus, these works can roughly be categorized into three groups. The first group concentrates on the input data of ML models to better understand the data distribution [9, 78] or to better select the high-dimensional data features [32, 48]. The second group focuses on the intermediate data representations from ML models to interpret how the data has been transformed internally. For example, most white-box interpretation solutions for deep learning models [57, 58, 66, 73] visualize the activations of different neurons from hidden layers, to reveal what have been captured/memorized by the neural networks. The third group targets the output from ML models to evaluate and compare different models. For example, *ModelTracker* [2] and *Squares* [60] use glyphs to encode the prediction probability of ML models and empower ML designers with instance-level data inspections and analysis. *MLCube* [30] allows users to compare ML models' performances (accuracies) over subsets defined using feature conditions. *Facets* [25] provides visualizations that aid in understanding ML datasets via individual feature exploration and subdivision of large data sets. To better disclose ML models' performance evolution, multiple visual designs for temporal confusion matrix have also been proposed, e.g., [28, 43].

Our work fits well with this third group, where we focus on analyzing and comparing the outputs from multiple debiasing techniques for word representations.

Finally, some existing work analyzes the visual analytics systems themselves, since biases may be introduced through the design decisions on interfaces and default parameters, e.g., [69, 70]. We, however, do not study this type of bias in this paper.

**Visualization for NLP.** Visualization has been employed for various NLP tasks such as topic modeling and sentiment analysis. For topic modeling, Chuang et al. [11] introduced *Termite* to visually assess topic model quality. Smith et al. [64] presented *Hiérarchie* that interactively visualizes large, hierarchical topic models. Liu et al. [44] created visual exploration to help users understand hierarchical topic evolution in text streams. For sentiment analysis, Smith et al. [63] further presented a so-called *relationship enriched visualization* that helps users explore topic models via corrections among words and topics. Wang et al. [71] introduced *SentiView* to analyze and visualize public sentiments of social media texts and their evolution. Liu et al. [42] used optimization to design *StoryFlow*, a storyline visualization system that illustrates the dynamic relations among entities in a given story. Liu et al. [39] introduced *NLIZE*, a visual analytic system that enables perturbation-driven exploration [38] of a natural language inference model, where a user can perturb a model's input, attention, and prediction.

Instead of focusing on abstract concepts such as topics and sentiments, our work aims to understand fine-level details presented in word embeddings, in particular, how the word embeddings are changed geometrically by various debiasing techniques. Word embeddings are considered as a type of word representation that allows words with similar meaning to have similar representations. Specifically, a word vector is a high-dimensional real-valued vector where semantically similar words have similar vectors. Several works in the literature are most relevant to ours in terms of visually exploring the space of word embeddings, see [26] for a survey of using visualization for interpreting word embeddings. Liu et al. [37] studied the pair-wise analogy relationships of word embeddings and proposed a new projection method to better preserve the analogy relationships in the projection space. Rathore et al. [57] visualized and investigated a graph-based summary of a collection of word embeddings obtained from the BERT (Bidirectional Encoder Representations from

Transformers) family of models [17]. Our tool is similar to [37], in that it enables the interrogation and interpretation via projected views of embeddings, but goes beyond in guiding the modification of the embeddings.

**Visualizing embeddings or latent spaces.** Word embeddings are a type of point cloud data for which generic high-dimensional visualization techniques may be applicable (see [40] for a survey). Since word embeddings are typically obtained via neural networks, techniques developed for visualizing latent spaces or hidden representations of neural networks are also relevant.

To visualize high-dimensional embeddings, dimensionality reduction algorithms (e.g., PCA, t-SNE [68], and UMAP [49, 50]) are commonly used in their analysis and visualization. Openly available toolkits such as *scikit-learn* [53] implements a number of such algorithms. Two analytical tasks are often focused upon. The *first* one is to interpret the semantics encoded in embeddings (e.g., [37]). For example, Smilkov et al. [62] developed *Embedding Projector* as part of the TensorFlow framework [1], enabling users to conveniently interact with embedding data and their local neighbors to quantitatively evaluate the embeddings. Rauber et al. [58] employed t-SNE projections to visualize the hidden representations of deep neural networks across neural layers, to reveal how data instances of the same class progressively form clusters. Multiple visualization efforts aimed to disentangle the latent space of deep generative models by relating the latent dimensions with human-understandable visual concepts [45, 46, 74]. The *second* analytical task is to compare embeddings generated from different algorithms. For example, *embeddingVis* [35] focuses on graph embeddings and uses multiple juxtaposed t-SNE views to compare different embedding methods. The same data instances are linked across all t-SNE views for explicit tracking. Heimerl et al. [27] proposed a set of metrics to measure the relationships (embedding correspondences) between two embeddings in a coordinated multiview system called *embComp*. Ghosh et al. [22] introduced a toolkit – *VisExPreS* – to disclose and compare the preserved global and local structures from the embeddings for novice data analysts. Kim et al. introduced *InterAxis* [31], which allows a user to interactively steer the scatterplot axes as linear combinations of data attributes.

Our work covers both analytical tasks. For semantic interpretation, we focus on the biases encoded in word embeddings and provide interactive applications that remove the biases through subspace transformations. For embedding comparison, we allow any embedding to be analyzed before and after dynamic modification, and to compare how different debiasing techniques affect its underlying geometry.

Our work also intersects with visualization for ML fairness (e.g., [7, 33]). It is specifically focusing on bias mitigation techniques in word embeddings. The *What-If* tool [24] combines data exploration with counterfactual (what if) explanations [33] and fairness modifications. *FairVis* [7] is used to audit pretrained models for biases against known vulnerable groups in the context of a recidivism prediction system.

The *DebIE* [20] platform, which offers a web interface for measuring and mitigating bias in word embeddings, and the *WordBias* [21] system which allows exploration of intersectional biases in static word embeddings are most relevant to our work. Both *DebIE* and VERB (which first appeared contemporaneously) illustrate bias in word representations, support the selection of certain debiasing algorithms, and utilize a collection of bias measures for evaluation. Compared with *DebIE*, the key additional contribution of VERB is that it provides a visualization interface that supports interactive exploration of the debiasing process and the affected embeddings. This interface in VERB highlights a common abstraction of these debiasing methods into a series of interchangeable components, and discloses the intermediate sequence of geometric transformations. While *DebIE* presents the end result of a debiasing algorithm, the step-by-step decomposition of the underlying process shown in VERB, we believe, is essential to make the identification and addressing of bias more accessible, to educate users of the debiasing techniques, and to allow practitioners to visually verify the effect of the

debiasing process. WordBias, on the other hand, focuses on visualizing bias in static word embeddings along multiple concept directions simultaneously, using a histogram of bias scores and a parallel coordinate plot of different types of bias. It allows users to add their own bias direction to enable exploration of intersectional biases. However, WordBias does not have the functionality to apply bias mitigation techniques, and thus lacks capabilities for modifying the embeddings to reduce the the identified biases. VERB closes the gap by visualizing geometric transformations of the embeddings to mitigate these biases, but limits the number of simultaneous biases that can be visualized. It would be interesting to integrate intersectional biases from WordBias to enhance VERB in the future.

VERB integrates different aspects of debiasing in word representations with several debiasing techniques, encompassing (a) bias subspace determination, (b) generalizable [14] bias mitigation strategies, and (c) bias measurement. It helps highlight and compare the different combinations of subspace identification and bias mitigation strategies that work best for a given embedding and a particular type of bias. VERB can also be utilized by users without previous knowledge or intuitions about potential biases to find issues within an incoming prediction model.

### 2.2 Bias in Word Representations

The Euclidean nature of embedded word representations makes them easy to integrate into a variety of NLP tasks and applications [52]. Furthermore, the similarity between the word representations is encoded in a way that respects their complex interactions and often takes the nuance out of modeling and formalizing those notions. However, embedded representations also come with challenges. Their *distributed* nature means that the original features in the text are no longer bound to specific dimensions. Therefore, their easy-to-interpret properties become obfuscated. A more troubling aspect is that these representations encode and potentially hide biases. Caliskan et al. [8] first revealed that these representations encode common stereotypes, where male identifiers are more associated with careers and female identifiers are more associated with families. In social studies of texts, as illustrated in Fig. 1, adjectives such as “rational” and “dependable” are often used to describe male leaders whereas “temperamental” and “excitable” are often used to describe female leaders. Such associations can be potentially harmful if these representations are used in tasks such as resume sorting, where female candidates can unwittingly be given lower rankings because of this hidden gender bias. Additionally, many such societal biases about different population attributes such as gender, race, nationality, ethnicity, age, and so on can be encoded [5, 8, 16] and unknowingly propagated downstream into tasks in NLP, resulting in harm [4, 13, 56, 67].

In order to mitigate such harm in diverse applications in which word representations are used, different debiasing strategies are being actively developed, the most common of which are based on postprocessing of the embedding spaces [5, 14, 16]. These methods are popular for their cost-effective approach and can be invoked on-demand depending on learning task at hand and extended to contextual embeddings [13]. We discuss these methods that we use in VERB in more detail in Sect. 4.

## 3 DESIGN CHALLENGES AND REQUIREMENTS

Our goal is to build an interactive visualization tool that increases the interpretability and accessibility of a set of state-of-the-art debiasing techniques. Specifically, we aim to help users better understand biases in word representations, and how various bias mitigation techniques work from a geometric perspective. By studying prior work, obtaining input from collaborating NLP and ML experts, we identify three design challenges (C1-C3) associated with developing our tool.

Prior to our work, existing debiasing pipelines had applied bias mitigation techniques and had assessed the outcomes via anecdotal evidence or numerical evaluations. The geometric transformation applied to word embeddings remained opaque to a user. Our tools aims to provide transparency to users in three key ways.

**C1: Identifying and visually disclosing biases.** Word vector representations are high-dimensional encodings of words that are not easily interpretable. Visualizing these representations and the biases encoded within them is nontrivial. Novices, and even some experts, need to be shown that biases indeed exist. Although helpful, simply applying dimensionality reduction techniques without careful considerations of viewing angles or distortions may not easily or faithfully reveal the existence or the direction of biases. As a result, users may easily miss or misinterpret biases in their data.

**C2: Intuitively demonstrating the debiasing process and visually verifying the effect of debiasing.** A primary rationale for building our tool is to provide easier access to a collection of common debiasing implementations and reduce the barrier of entry for applying these techniques. The target audience of our tool is AI and ML experts who design and implement a data analysis pipeline. At the same time, the tool should also be suitable as a pedagogical tool for explaining the main concepts to non-experts with little NLP background to encourage them to explore these bias issues in more depth. Most prior work presents only the end-to-end results, without disclosing any intermediate details or interpreting the debiasing procedures.

**C3: Establishing a common ground that bridges debiasing with visualization and enables human-in-the-loop modifications of embeddings.** As we have reviewed in Sect. 2, there are a number of outstanding visualization techniques, impressive visual analytics frameworks, and prevalents debiasing algorithms. However, most of these works are conducted in their respective fields in isolation. There are missed opportunities in building a general pipeline that connects these research outcomes, where users perform human-in-the-loop investigations and modifications of high-dimensional embeddings. In particular, users can use visualization to modify and diagnose the word representations before and after debiasing, and reuse the visual analytic results for downstream tasks.

VERB has been developed in close collaboration with ML experts who study biases in embeddings. Among them, two experts have coauthored this work, and they are also the inventors of multiple debiasing algorithms [12, 14, 16] that are part of the backend of VERB (Sect. 4). Based on our interactions with ML experts, attendee feedback collected from an initial version used at two AI conference tutorials (AAAI and KDD), and literature reviews, we distill the above design challenges to the following design requirements (R1-R3) that guide VERB's development:

**R1: Demonstrating the existence of biases through a visual interface that preserves geometric intuitions and known linear structures.** Dimensionality reduction algorithms are good candidates for addressing C1; however, our tool will need to ensure the biases are not distorted in the process. As will be elaborated on in Sect. 4, the debiasing algorithms are inherently geometric and rely on linear subspaces. Hence, we require that our visualizations are also geometric in nature and preserve and highlight the linear structures of the biases utilized in the debiasing algorithms. Yet, they should not hide the nuanced, partially interchangeable, and noisy nature of these approaches.

**R2: Providing a modular decomposition that allows users to comprehend and explore the commonalities and distinctions between common debiasing methods.** To address C2, our tool will explain how to use debiasing approaches, the subtle but meaningful differences between them, what their constraints and limitations are, and the benefits they can deliver. We will decompose the well-encapsulated debiasing process into key components and visualize the intermediate representations of the embeddings for better interpretation. Specifically, as explained in Sect. 4, each debiasing approach is decomposed into three modular components: (1) subspace identification, (2) embedding modification, and (3) bias

evaluation. Such a decomposition allows for a general and interchangeable set of techniques. These three components should be individually demonstrated, and users should be able to choose how to combine them.

For the first component, the identified subspaces can and should help ground the correct reference frame to comprehend the modifications to the underlying representations. For the second component, the intermediate transformations and transitions between them should use smooth transitions so users can preserve their mental map in a continuous fashion, while observing these already complicated transformations. For the third component, intrinsic and extrinsic evaluation measures help users determine how effective the bias removal has been.

**R3: Integrating with data analysis pipelines.** Our tool will allow users to visually verify the effect of debiasing as part of a larger data analysis pipeline. The debiasing methods' output is also a word embedding, and users might want to perform downstream evaluations and tasks. Users may also want to use additional tools to perform visual analytics on the transformed embeddings. The tool may be used as a standalone analysis tool or integrated into a larger analysis pipeline. When used as part of a larger analysis, the tool should be used to visually verify the effectiveness of the chosen debiasing mechanism and seed words in transforming a representative set of evaluation words. Moreover, the tool should have the ability to export the debiased embedding in an easily parsable and commonly used format to be used in the later parts of the pipeline.

**VERB system overview.** Adhering to these design requirements, we next describe the design and implementation of the VERB system. The VERB backend is composed of multiple debiasing methods. These methods, and how to evaluate them, are described in Sect. 4. We demonstrate the usefulness of VERB by showing snapshots of intermediated embeddings to explain how these methods work. The VERB frontend is detailed in Sect. 5. Its interface consists of a Control Panel that enables the configuration of a particular debiasing algorithm, an Embedding View that highlights a subset of word embeddings and visualizes their step-by-step transformations across various debiasing techniques, and an Explanation Panel that gives a step-by-step description of the transformation, as illustrated in Fig. 1.

## 4 VERB BACKEND: DEBIASING EMBEDDED REPRESENTATIONS

In this section, we give an overview of four debiasing methods for embedded representations (e.g., [51]), which form the backend of VERB. These methods are becoming essential elements of many pipelines to process and understand texts and other types of complex data (e.g., merchant embeddings in Sect. 6). In particular, we leverage snapshots from VERB to provide an intuitive explanation of the main components of these methods.

In developing VERB, we identify that all of this class of mechanisms can be decoupled into a three-step process. The first is to identify a linear concept subspace among the vectorized representations that capture the direction of bias (e.g., the concept of gender or nationality). The second is to use this subspace to transform the representations in a simple and controlled way. The last is to evaluate the transformed representations.

This decoupling process provides several important advantages for VERB in addressing our design requirements. First, by identifying the linear subspaces, it orients the embedding view to show the more relevant, and only modified components, in accordance with ⟨R1⟩. This way, users can verify how changes are happening with respect to concepts and not worry that other unseen dimensions are being distorted. Second, it allows the explanation of the debiasing process to be presented in simpler and easier to digest components. Third, it allows VERB to be modular, and allows users to mix-and-match these components and find effective ways to pair them for the task at hand, addressing ⟨R2⟩.

### 4.1 Step 1: Subspace Identification

In embedded representations, the specific dimensions occupied by features are unknown. In this section, we discuss four methods in the literature used to determine the subspace that is the span of a specific concept (e.g., gender). Some of these methods (PCA and PCA-paired) naturally generalize to identify multiple directions, but it is quite rare to use more than one direction to represent a concept. To keep subspace identification modular and simple, VERB currently identifies 1-dimensional subspaces, as described below. The identified direction is independent from any subsequent visualization or debiasing mechanism.

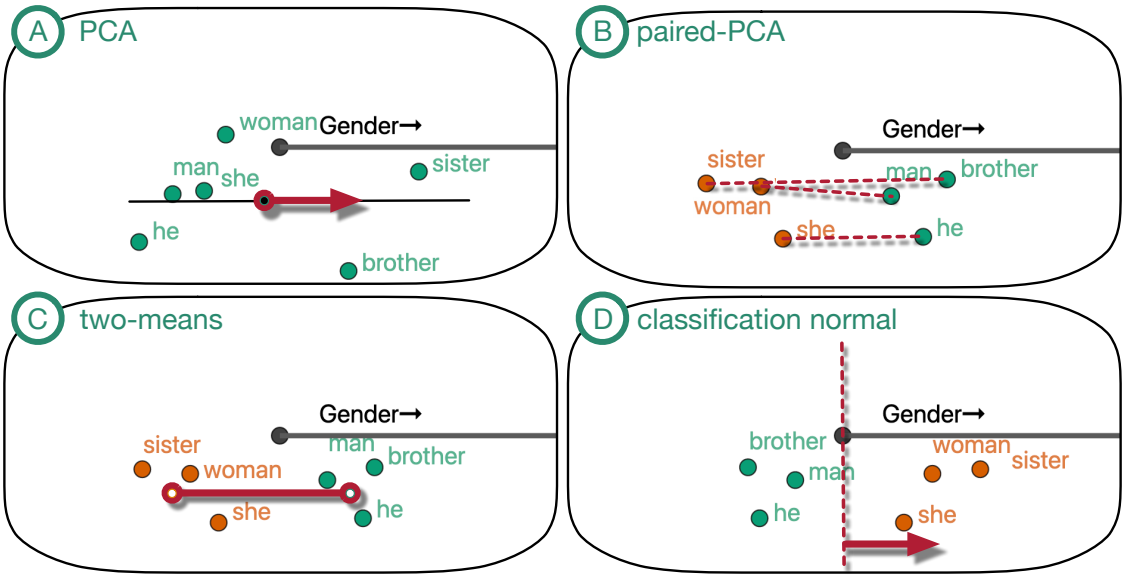


Fig. 2. Concrete examples of subspace identification methods using VERB. (A-D): PCA, paired-PCA, 2-means, and classification normal. (A): green points are seed words. (B-D): green points represent male gendered words, orange points represent female gendered words. The black line segment that starts from the origin represents the gender subspace direction.

**PCA.** This is a general and simple approach to determine a subspace. It requires one set of word vectors, referred to as the *seed words*, from which it computes the top principal component – which is the best 1-dimensional subspace that minimizes the sum of squared distances from all word vectors. This resulting unit vector represents the subspace direction. Using VERB, we illustrate the PCA method in Fig. 2A using a set of gendered seed words:"man, woman, brother, sister, he, she". The red arrow above the black line shows the direction of the gender subspace obtained via PCA.

**Paired-PCA.** Another variant based on PCA was proposed by Bolukbasi et al. [5]. It requires a list of paired words as the seeds; each pair has one word vector from different groups. For example, for the gender concept shown in Fig. 2B, we use "man-woman, he-she, brother-sister" as seeds for subspace identification. The paired-PCA method then reports the concept subspace as the first principle component of the difference vectors between each paired vectors. Because these vectors are the result of differences, we do not need to "center" them (remove their mean) first as when PCA is used on word vectors.

**2-Means method.** The 2-means method [16], for any two sets of words as seed sets, returns the normalized difference vector of their respective averages. So for groups of words $F = \{f_i\}$ and $M = \{m_i\}$, it computes $f = \frac{1}{|F|}\sum_i f_i$ and $m = \frac{1}{|M|}\sum_i m_i$ as the mean of each set. Then the direction is calculated as $v = \frac{f-m}{\|f-m\|}$. This method has the advantage that it does not require paired words or an equal number of words in the two seed sets. We give an example of applying the 2-means method to two sets of seed words in Fig. 2C, where $F = \{\text{“}woman'', \text{“}sister'', \text{“}she''\}$ and $M = \{\text{“}man'', \text{“}brother'', \text{“}he''\}$. The computed gender direction (black line segment) originates from the origin in the visualization.

**Classification normal.** For two groups of seed words that can be classified using a linear support vector machine (SVM), the direction perpendicular to the classification boundary represents the direction of the difference between the two sets. Again, this requires only two sets $F$ and $M$, but they do not need to be paired or of equal size. As illustrated in Fig. 2D, the dotted line represents the classification boundary between $F = \{\text{“}woman'', \text{“}sister'', \text{“}she''\}$ and $M = \{\text{“}man'', \text{“}brother'', \text{“}he''\}$, and the red arrow is its normal direction. The black segment emanating from the origin again indicates the gender direction. Ravfogel et al. [59] used this direction iteratively to remove bias in word vectors by projections.

### 4.2 Step 2: Bias Mitigation

There are several methods to modify the embedding structure in ways that mitigate the encoded bias. Although there are more complicated optimization-based ones designed for specific tasks in gender bias in text [82], we describe a subset of four debiasing methods that are quite simple to actuate (although nuances of them may be intricate), and they rely specifically on the concept subspaces identified earlier. Again, VERB serves as the perfect visual medium to explain these debiasing methods. For the descriptions below, a point in the space of high-dimensional embedded representations is denoted as $x \in \mathbb{R}^d$ (e.g. for $d = 50$ or $d = 300$). A concept subspace is labeled $v$ and is restricted to be a unit vector in $\mathbb{R}^d$.

**Linear Projection (LP).** This approach [16] removes the component of concept subspace for each data point $x$. This procedure can be applied individually to each data point $x$, where the component along $v$ is $\langle v, x\rangle v$, where $\langle v, x\rangle$ is the Euclidean dot product. The LP method then removes the component along $v$ for every point $x \in \mathbb{R}^d$ as $x' = x - \langle v, x\rangle v$.

Using VERB, we give a simple example by applying two-means and LP debiasing in mitigating the gender bias in occupational words. The two seed sets are $M$={“man”, “he”} and $F$={“woman”, “she”}. The evaluation set is $E$ = {“receptionist”, “nurse”, “scientist”, “mathematician”}. As illustrated in Fig. 3, VERB decomposes the LP method into an interpretable sequence of transformations. In Step 0, both seed sets and evaluation set are viewed using a perspective from PCA, where the gender direction is identified using two-means. In Step 1, the viewing perspective/angle is reoriented so the gender direction is aligned with the x-axis, where we see clearly that “receptionist” and “nurse” are shown to be closer to the female direction whereas “banker” and “engineer” are closer to the male direction. The reorientation does not change any data, it simply changes the 2-dimensional subspace that is visualized. The VERB interface smoothly animates this by interpolating the viewing angles. In Step 2, for every word in the embedding, LP removes its component along the gender direction in $\mathbb{R}^d$, where all words are shown to be aligned along the vertical axis. The underlying data is modified in this step. In Step 3, the transformed (debiased) points are reoriented again using the perspective from PCA, where there is no clear gender association among the occupational words. This last PCA view is different from the original PCA view since the data was modified in Step 2.

**Hard Debiasing (HD).** An earlier approach (the first one proposed) by Bolukbasi et al. [5], known as Hard Debiasing, uses a similar mechanism, and is designed specifically for gender bias. It also requires an additional wordlist called the

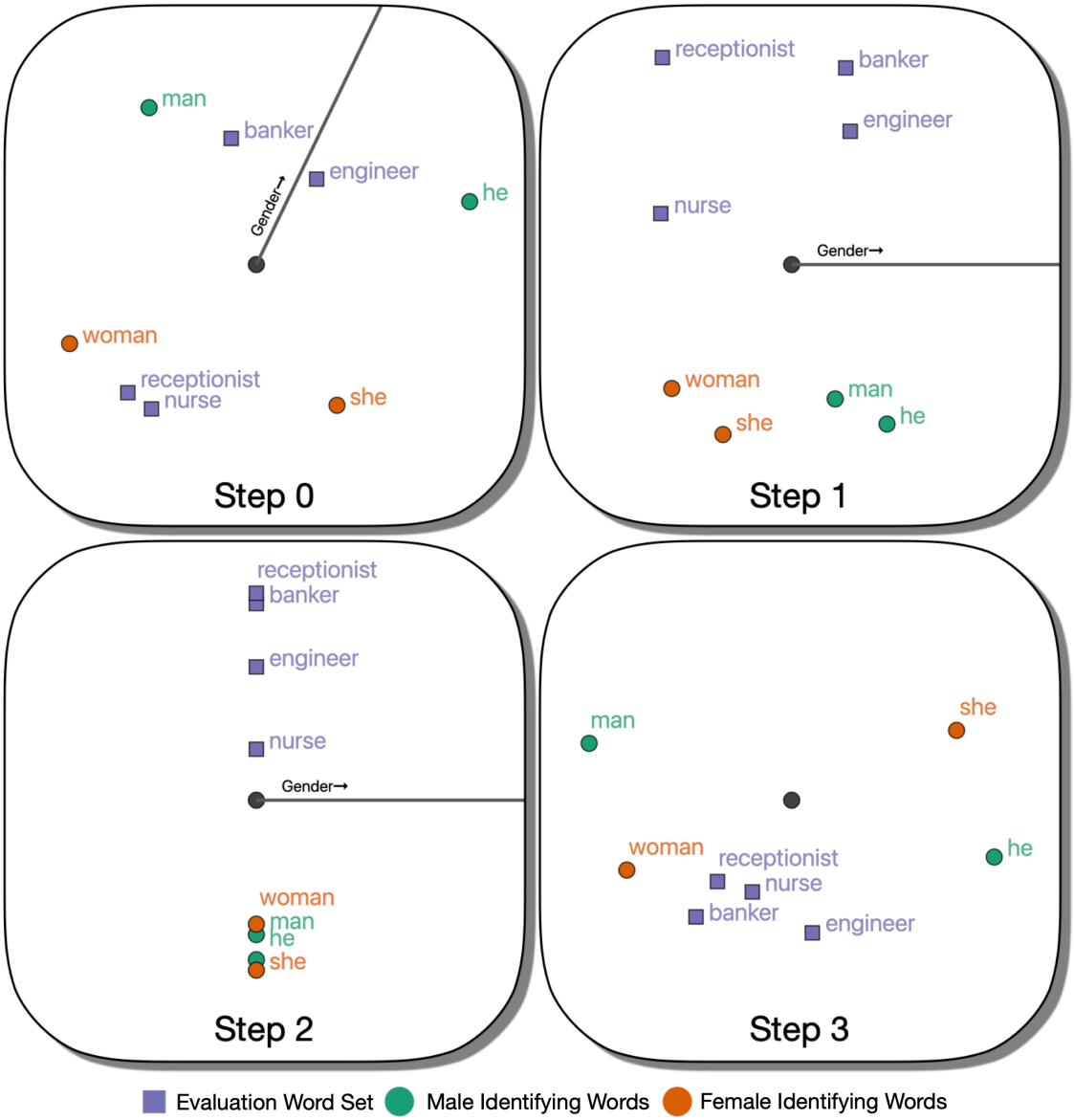


Fig. 3. A simple example using two-means to identify a gender subspace and Linear Projection to mitigate the gender bias in word embeddings.

*equalize set*, which is used to preserve some of the information about that concept. We summarize this mechanism next. The words that are used to define $v$ are considered definitionally gendered and not modified. The exception is another provided set of pairs of words (e.g., "boy-girl", "man-woman", "dad-mom", "brother-sister"). These word pairs are *equalized*; that is, they are first projected as in Linear Projection, but then each pair is extended along the direction $v$, so

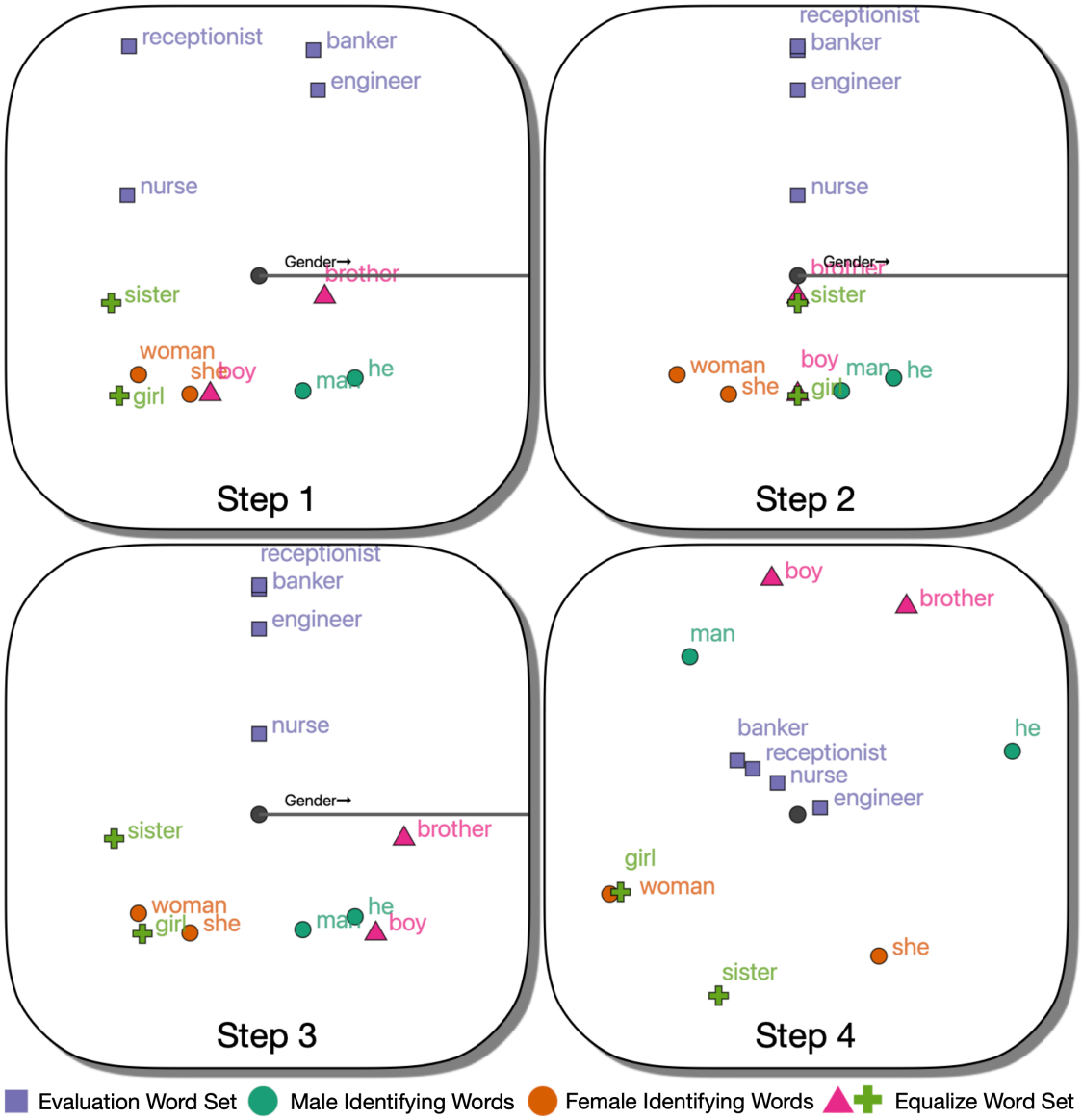


Fig. 4. An example using two-means and HD to mitigate the gender bias.

the words are equally far apart as they were before the operation. The remaining words are then projected as in Linear Projection.

In our example with VERB, we again use $M$={"he, man"} and $F$ ={"she, woman"} and two-means to define a gender direction $v$, $Q$= {"boy-girl","sister-brother"} as the equalize set, and $E$= {"engineer", "lawyer", "receptionist", "nurse" } again as the evaluation set. As illustrated in Fig. 4, Step 1 is obtained after a reorientation of the gender direction along the x-axis. Step 2 is removing the component of each point along the gender direction with the exception of $M$ and $F$

("she, woman" and "he, man"). Step 3 tries to preserve some information regarding gender using the equalize set $E$ thus extending the words in $Q$ ("brother," "sister," "boy," "girl") along the gender direction so they become equally far apart. Step 4 reorients the modified words using PCA from a viewing perspective with the most variance.

Bolukasi et al. [5] described other methods, and later works by Wang et al. [75] also provided slight variants, or rediscovered these approaches. One concern about Hard Debiasing is that it may leave residual bias [23]. The authors of that critique helped develop the next approach as an alternative.

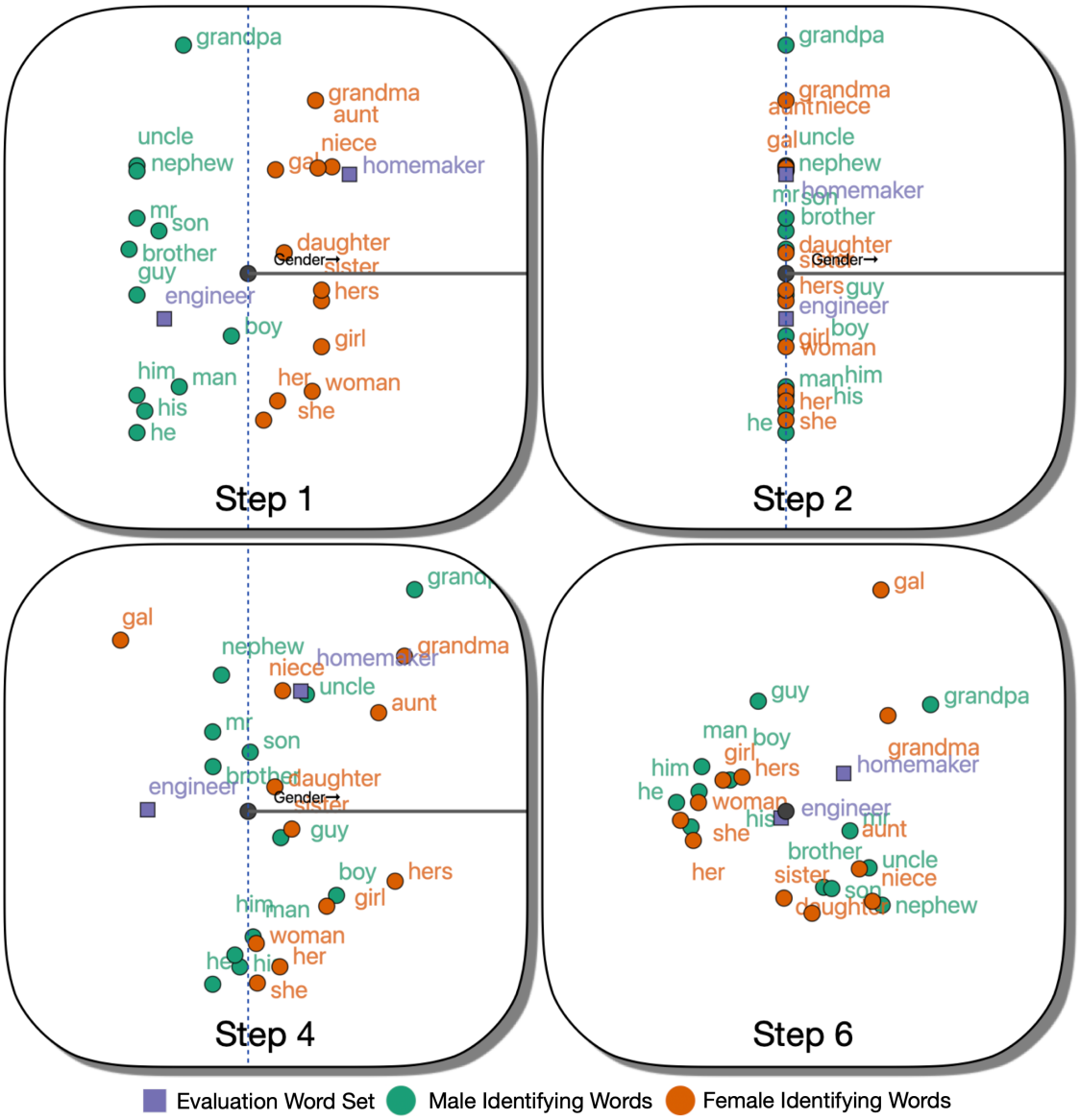


Fig. 5. An example using classifier normal and INLP to mitigate the gender bias over two rounds.

**Iterative Nullspace Projection (INLP).** INLP [59] starts with a pair of large word lists (e.g., sets of male and female words). It suggests to select the top 0.5% of the extreme words along either directions of the he-she vector, denoted as sets $M$ and $F$, respectively. It then builds a linear classifier that best separates $M$ and $F$, and linearly projects all words along the classifier normal (denoted as $v_1$). However, a classifier with accuracy better than random may still be built on $M$ and $F$ after the projection. Let $v_2$ denote the classifier normal. INLP then applies linear projection to all words again along $v_2$. This continues for some large number of iterations or when no better classifier can be found. Afterwards, the words that may encode bias, even by association (the sets $M$ and $F$), cannot be linearly separated with accuracy better than random chance.

An example run of INLP using VERB is shown in Fig. 5 using two sets of definitionally gendered words $M$ = {"man, he, him, his, guy, boy, grandpa, uncle, brother, son, nephew, mr"} and $F$ = {"woman, she, her, hers, gal, girl, grandma, aunt, sister, daughter, niece"}. A perfect separator/classifier can be found initially (shown in Step 1), and then linear projection along the classifier normal is shown in Step 2. The next classifier normal (shown in Step 4) is not a perfect separator. Yet, after its next application, and a PCA reorientation as shown in Step 6, no sufficiently good classifier can be found, and the procedure stops.

**Orthogonal Subspace Correction and Rectification (OSCaR).** A critique of the above techniques, especially INLP, is that they are destroying information that we might want to preserve. For example, we may want to know that "grandpa" is referring to a male grandparent. The OSCaR algorithm [14] seeks a more controlled approach. It requires two specific concept subspaces, for instance, one representing gender $v_1$ and another representing occupations $v_2$. OSCaR does not "project away" the gender subspace, but rather attempts to disassociate them by making those subspaces *orthogonal*. In addition to orthogonalizing those subspaces, which can be done by rotating $v_2$ to $v_2'$ so $\langle v_1, v_2' \rangle = 0$, it also rotates all other data points by a lesser amount. Points close to $v_1$ do not rotate much, whereas points close to $v_2$ rotate about as much as $v_2$. Whereas OSCaR does not remove any possible way to find any association between data aligned with either of these subspaces, it does make the concepts as a whole orthogonal. In the bias evaluation approaches described in Sect. 4.3, OSCaR is demonstrated to reduce bias in an amount similar to other debiasing approaches. Moreover, it retains the information along each of the original subspaces $v_1$ and $v_2$.

With VERB, Fig. 6 shows the four steps of OSCaR. The first subspace $v_1$ representing gender is defined with words "he," "his," "him," "she," "her," "hers," "man," and "woman." The second subspace $v_2$ representing occupations is defined with words "engineer," "scientist," "lawyer," "banker," "nurse," "homemaker," "maid," and "receptionist." In the PCA view (Step 0), one can observe that the two subspaces are correlated, and the typical gender stereotypes of the occupation are present in the word representation, e.g., "maid" in the female direction and "engineer" in the male direction. The reoriented view in Step 1 aligns the Gender direction ($v_1$) along the x-axis. It shows the span of $v_1$ and $v_2$, which is the 2-dimensional subspace where OSCaR modifies the data. It is also the subspace with the largest angle between these two subspaces. In Step 2, the data is modified so that the gender and occupation subspaces become orthogonal. The evaluation set words "grandma," "grandpa," and "programmer" (along with all other words) can be seen to move along with these words. Note how "programmer" is still near the other technical-oriented careers, and how "grandpa-grandma" retains the inherently male-female relationship. Finally, in Step 3, another PCA view is shown on the modified data, and now the subspaces can be seen to retain the orthogonal nature, and the gender connotation in the occupations has been rectified, so there is no apparent stereotypical correlation.

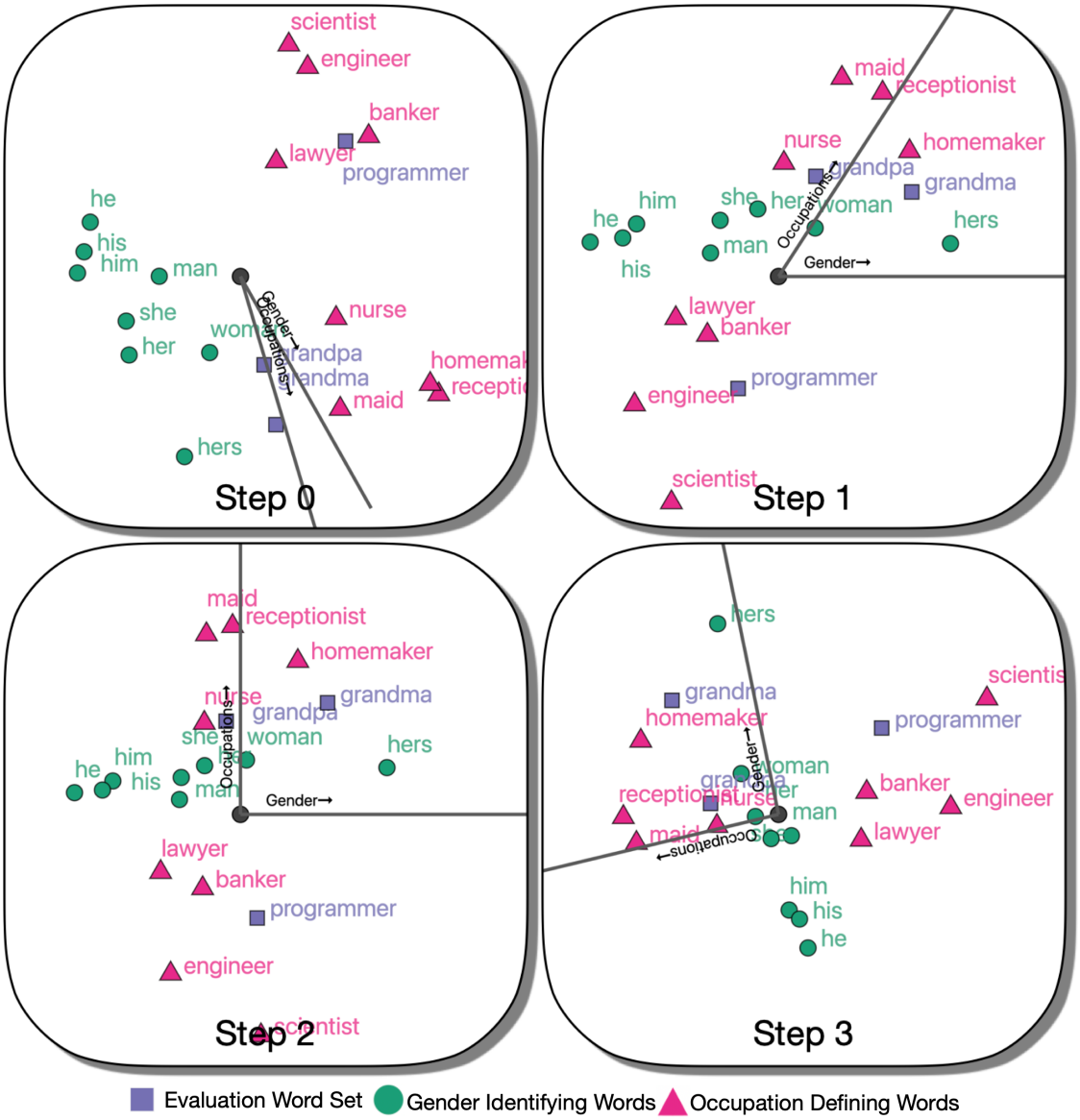


Fig. 6. An example using PCA and OSCaR to rectify gender bias in relation to occupations.

### 4.3 Step 3: Bias Evaluation

There are several intrinsic [8, 16] and extrinsic [13, 82] measures to determine how much bias is contained by word embeddings. When bias is removed [5, 13, 14], these measures help determine how effective the bias removal has been. In general, it may not be possible to completely remove bias in these measures due to the nature of the measurement or its influence from other data and training choices. We next describe some common and representative bias measurement methods.

*4.3.1 WEAT.* The Word Embedding Association Test (WEAT) [8] is an analogue to the Implicit Association Test (IAT) from psychology. It checks for human-like bias associated with words in word embeddings. For example, it finds career-oriented words (e.g., "executive", "career") are more associated with statistically male names (e.g., "Tom", "Peter") and male-gendered words (e.g., "man", "boy"), whereas family-oriented words (e.g., "family", "home") are more associated with statistically female names (e.g., "Mary", "Kate") and female-gendered words (e.g., "women", "girl").

WEAT considers four sets of words: two target word sets $X$ and $Y$ (e.g., representing male and female genders) and two sets of attribute words $A$ and $B$ (e.g., representing stereotypical male or female professions). First, for each target word $w \in X \cup Y$, it computes how much the word is associated with set $A$, and not associated with set $B$ as

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

where $\cos(a, w)$ is the cosine similarity between vector $a$ and $w$. Then it averages $s(w, A, B)$ across all $w \in X$, minus the average of all $w \in Y$ as

$$s(X, Y, A, B) = \frac{1}{|X|} \sum_{x \in X} s(x, A, B) - \frac{1}{|Y|} \sum_{y \in Y} s(y, A, B).$$

Finally, the WEAT test statistic is $s(X, Y, A, B)$ normalized by the standard deviation of $s(w, A, B)$ for all $w \in X \cup Y$, so typical values should not be too far from $[-1, 1]$, and a value closer to 0 indicates less implicit (and biased) association.

VERB allows users to compute WEAT before and after debiasing. The default word sets (which can be modified) in VERB use the following [8, 16] configuration:

- Male words as X = {male, man, boy, brother, he, him, his, son}
- Female words as Y = {female, woman, girl, sister, she, her, hers, daughter}
- Stereotypically male occupations A = {doctor, engineer, lawyer, mathematician, banker}
- Stereotypically female occupations B = {homemaker, receptionist, dancer, maid, nurse}

*4.3.2 Embedding Coherence Test.* The Embedding Coherence Test (ECT) [16] measures if groups of words have stereotypical associations. Instead of evaluating the exact word similarities (e.g., male and female words with occupation words), it first aggregates the male and female words into their means, described as $m = \frac{1}{|X|} \sum_{x \in X} x$ and $f = \frac{1}{|Y|} \sum_{y \in Y} y$. Then, it evaluates the order of similarity from $m$ and from $f$ to a different set $A \cup B$, such as occupation words ("doctor", "nurse", etc.). Next, it sorts the values $\cos(m, w)$ for each $w \in A \cup B$ and the values $\cos(f, w)$. The similarity of these sorted lists is measured with the Spearman Coefficient, which ranges between 1 (when the ordering is exactly the same, so the least bias) and -1 (where the ordering is exactly opposite, thus the most biased). Therefore, larger values of ECT indicates less bias.

*4.3.3 NLI Based Tests.* Since word representations are used downstream in different tasks and applications in NLP, it is important to measure the effect biased associations have on the decisions made in these tasks. An example is natural language inference (NLI). Dev et al. [13] used NLI to provide a clear signal on the encoded bias. The task is, given a pair of sentences to predict if the second one is entailed, is contradicted, or is neutral to the first sentence. The sentence pairs constructed as all neutral and any deviation from a neutral prediction is bias. These are constructed from simple template sentences where the VERB, object, and subject are chosen from word lists, and in total over a million sentences are considered. For each one, the subject (e.g., an occupation like "doctor") is paired with another sentence where the subject is replaced by either "man" or by "woman." If the occupation has no gender bias, it will result in a neutral

inference prediction, but if bias is encoded, it will result in higher probability of entailment or contradiction. The higher the percentage of sentences predicted as neutral, the better the prediction (i.e., the lower the amount of bias).

Currently, VERB supports WEAT-based evaluation in its visual interface due to its simplicity in computation. We use all three measures to evaluate a new, VERB-inspired subspace identification method in Sect. 6.3.2.

## 5 VERB FRONTEND: USER INTERFACE

With VERB, users can explore and interpret four types of debiasing techniques (see Sect. 4.2) through three coordinated views.

The **Embedding View** (Fig. 1A) highlights a subset of word vectors using dimensionality reduction and visualizes the transformations of their embeddings across various debiasing techniques. It decomposes a chosen technique into a sequence of interpretable operations and visualizes their associated transformations via a step-by-step animation, in accordance with ⟨R2⟩. It also provides additional capabilities to interact with individual word vectors. In particular, users can select a word in the Embedding View, and VERB will display its nearest neighbors in the high-dimensional embedding space, before and after debiasing. Users can hover over a word and there is a popup to display quantitative association of the word vector with the bias subspace/direction (using dot product between the word vector and the susbspace direction) before and after debiasing.

The **Control Panel** (Fig. 1B) enables users to configure each debiasing technique by specifying the algorithm (LP, HD, INLP, or OSCaR), subspace technique (PCA, paired-PCA, 2-means, or classification normal), concept labels (e.g., gender, occupation), seed sets (for defining concept directions), evaluation set, and equalize set, etc. The evaluation set does not influence the debiasing methods; rather, it allows users to observe how the methods affect any chosen words contributing to ⟨R3⟩. Similarly, modifying the seed sets allows users to see how robust the concept subspaces and debiasing mechanisms are to those choices – an essential property to ensure robustness in a larger data analysis pipeline. The Control Panel also provides controls to navigate through the steps of the chosen technique and to toggle various aspects of the visualization such as data labels, subspace direction, and evaluation points, enabling users to construct mental models of the process's minutiae as emphasized in ⟨R2⟩. Users can also choose from a list of predefined examples (detailed in Sect. 6). The Control Panel also provides users the ability to export the debiased embeddings, addressing ⟨R3⟩.

The **Explanation Panel** (Fig. 1C) gives a step-by-step description of the transformation. VERB also enables users to download the modified word embedding after applying a particular debiasing technique. Thus, VERB not only provides an educational guide for understanding a debiasing technique, but also allows users to apply and visually verify these modifications before moving to any downstream analysis.

These capabilities make VERB not just a visualization tool but also a component of data science pipelines that involve distributed representations, to address ⟨R3⟩. Due to fundamental differences in the number and types of inputs necessary for the various debiasing methods, it is hard to allow direct comparison inside the interface, and we defer this task to downstream components of the users' analysis pipeline. Users can also compute and compare standardized WEAT scores for the initial and debiased embedding right inside the interface, enabling them to perform a preliminary assessment of debiasing under the current set of chosen parameters. We note that optimizing for WEAT and other quantitative measures directly on the data used to define concepts could lead to overfitting. These measures need to be tuned for the concepts of interest, and that, in general, many of these scores are designed only for binary biases. Since significant caution should be used in interpreting the quantitative scores, beyond the standard gender-focused measure we provide within VERB, we advocate a deeper quantitative analysis be performed by experts downstream.

In the example shown in Fig. 1, VERB helps reveal the association in the word embedding with the *gender* concept that contributes to its gender bias. Specifically, before debiasing (Fig. 1A), along the gender direction (a black line segment starting from the origin), the adjectives "strong," "important," "arrogant," and "rational" are more closely associated with the male words "he" and "him", whereas the words "temperamental," "gossip," "excitable," and "beautiful" are more closely associated with the female words "she" and "her". VERB then animates the step-by-step transformation of the word embedding using LP for debiasing and two-means for subspace identification. After debiasing (Fig. 1D), there is no longer a gender direction, and the above adjectives do not show a clear gender bias.

**Transforming embedding views.** A central functionality of VERB is that it allows users to experiment with and visualize the effects associated with a chosen debiasing mechanism. Specifically, the tool updates the Embedding View when an algorithm modifies the underlying representation step-by-step, addressing ⟨R1⟩. We have experimented with multiple embedding views corresponding to major steps of the process, but have found that it required larger effort from users to track changes across multiple views. Having smooth transitions between steps helps users focus on the differences over steps instead of the word positions themselves. As we demonstrate in Sect. 6, VERB can be applied to not only word vector embeddings that arise from NLP, but also other embedded representations. Whereas the data is represented as 50 or often higher dimensional vectors, VERB provides views of the data objects in a 2-dimensional interface as points. Whereas our (default) underlying embedding has 100K points (others could have much more), we do not attempt to visualize all points. Instead, the Control Panel allows users to select a representative subset (i.e., the evaluation set) to visualize.

After choosing a debiasing mechanism and a subspace identification technique in the Control Panel, each debiasing process always starts with a 2-dimensional scatterplot, obtained using the best 2-dimensional subspace as determined by PCA on the user-provided data points. We choose PCA for the projections since it is a *linear* projection method that is widely used and understood. Any projection to a low-dimensional space by nature is losing information; without knowledge of the concept direction, the PCA projection loses the least information among linear projections. Debiasing being an inherently geometric operation, nonlinear projections such as t-SNE or UMAP might introduce distortions not present in the data, which goes against the design requirement ⟨R1⟩. Previous studies, such as *InterAxis* [31], have also proposed methods to visualize high-dimensional datasets in a 2-dimensional scatterplot using a user-centric approach for feature selection and weighing. For our purposes of analyzing word representations, where individual features (1) do not have physical grounding (the representations are *distributed*) and (2) interact in complex ways, *InterAxis* is not a suitable choice.

For the PCA projection, the origin is always shown in the center of the Embedding View. Since the data points are often interpreted as vectors, the cosine metric is the most commonly used metric (which measures angles with respect to the origin as a base point). This initial PCA view (marked as "Step 0" in the Embedding View) itself is not especially meaningful, but it is useful as an alternative starting point, and helps highlight the meaningfulness of the other views.

In "Step 1" of the transformation, VERB changes the viewing perspective of the initial embedded view (Fig. 1A). In particular, the concept subspace $v$ is always rotated so it is aligned with the $x$-axis. The $y$-axis is chosen as the highest variance direction among the remaining points (via PCA). This choice of $x$-axis is essential for two reasons from ⟨R1⟩. First, the left-right direction provides a faithful account of how far each representative point is along this concept subspace. Two related terms (e.g., "temperamental", "rational") can be compared, and their relation along this concept subspace (e.g., Gender) is not distorted, which may be the case if that subspace was not parallel with the viewing plane. Second, when a projection operation (internal to three of the debiasing mechanisms) is applied that effectively removes the component

along this concept subspace, then one can clearly see the representative data points moving onto a lower dimensional subspace combined into and represented by the $y$-axis.

The OSCaR mechanism requires the definition of two subspaces $v_1$ and $v_2$. In this case, the initial embedding view is the span of these two subspaces. In OSCaR, all of the operations happen only in this subspace, while all components outside this 2-dimensional subspace (e.g., coming out of, or into the screen) are not modified. As with all techniques, OSCaR allows users to explicitly see the action happening without any visual side effects that obscure these operations.

For the INLP operation, which iteratively applies linear projection after finding the new best classsifier, VERB shows each of these steps. After each new normal direction is found for a subspace, it updates the viewing perspective to make that subspace along the $x$-axis as before, so that residual concept can be viewed, and its projection can be dynamically visualized.

Finally, with the new (debiased) embedded representations, VERB provides a final view of the data from the 2-dimensional PCA perspective. This is important to show the final and best possible view of the representations, especially when a projection mechanism is used, otherwise, all of the data may have been compressed into a single 1-dimensional subspace (the $y$-axis) within the earlier perspective.

**Implementation.** The front-end of VERB is implemented using the HTML/CSS/JavaScript stack and D3.js. We use an automatic label placement algorithm [72] that uses simulated annealing to minimize overlaps between text labels in the Embedding View. Its back-end is developed using Python and Flask. VERB comes pre-loaded with a 50-dimensional GloVe embeddings of the 100K most frequent words taken from the Wikipedia 2014 + Gigaword corpus [54]. It also provides a larger downloadable GLoVe embedding (300-dimensional with the 100K most frequent words) from the Common Crawl corpus (https://commoncrawl.org/).

**Scalability.** The backend of VERB is designed with scalability in mind and has been tested to apply debiasing techniques to 300K word embeddings. The default setting applies debiasing techniques to a set of 100K GLoVe embeddings. These debiased embeddings can be downloaded by users using the "Export as CSV" feature in the tool. The frontend can display up to 10K words simultaneously, however, at the expense of increased visual clutter. The recommended usage of VERB is to perform visual exploration and analysis with dozens of words which most clearly define a concept (perhaps fewer in a pedagogical setting), and export the full debiased word embeddings for downstream analysis.

## 6 USE CASES

We evaluate the efficacy of VERB through multiple case studies, conducted with the targeted users of this tool. The studies have been carried out through guided explorations in three steps. First, we explain the high-level design goals/rationales of the system, along with necessary background of the debiasing algorithms. The users can then freely interact with the system to explore different embedding data and debiasing algorithms. The users work through their own explorations and there is no collaboration between users in this process. Finally, we collect feedback from the users through open-ended interviews and think-aloud discussions.

**Studied Cases.** This section presents the four case studies that we designed to showcase the capability of VERB and help users get familiar with the tool. First, we will show how VERB can be used to quickly and easily identify new forms of bias in word vector embeddings. Second, we will demonstrate the power of VERB in teaching and contrasting methods to identify concept subspaces and use them to attenuate bias in word vector embeddings. Third, we will highlight how VERB identifies concept subspaces as a critical yet under-explored element of debiasing. Inspired by this, we will show how to optimize subspace identification, leading to an improved iterative method which quantitatively improves the

debiasing results. Finally, we will showcase VERB's generality by exploring a different type of embedded representation, one that captures merchant embeddings associated with a large payment company.

**Embedding Datasets.** Two embedding datasets are involved in the studies. The word vector embeddings use the pretrained GloVe embeddings trained on the Wikipedia 2014 + Gigaword 5 corpus [54]. The merchant vectors are trained on financial transaction data (see [18], [79] for details).

**Users and Their Background.** Four groups of users have participated in our studies. The *first* group includes three workshop attendees ($W_1$~$W_3$) from the AAAI conference and five ($W_4$~$W_8$) from the KDD conference (where we presented tutorials of the system). The *second* group consists of two graduate students ($S_1$~$S_2$), majoring in computer science. Their research focuses on studying ML and NLP techniques, where exploring embedding data is an important part of their daily work. The *third* group includes two researchers ($T_1$~$T_2$) of an industrial research lab who work with the merchant embedding data everyday. The *fourth* group consists of 20+ researchers and Ph.D. students with varying backgrounds on data bias and model fairness analysis. The first three groups are considered experts, while the final group consists of a mixture of experts and non-experts. Different groups of users have their respective focuses of use cases based on their interest and familiarity with the explored datasets. Specifically, group 1 ($W_1$~$W_8$) and group 2 ($S_1$~$S_2$) explored the word embedding cases. Group 3 ($T_1$~$T_2$) explored all four cases but focused on the merchant embeddings, as it is their domain problem and they are very familiar with the dataset. The participants in group 4 were guided to explore all four cases. However, most of them focused on the cases with the word embedding dataset, as the dataset is easier for them to understand. The feedback from all users are presented later in Sect. 7.

### 6.1 Identifying New Types of Biases

VERB allows users to load any word vector embeddings or embedded representations (e.g., GloVe embeddings and merchant embeddings). Users can select any subset of words or identifiers (e.g., evaluation set) to quickly illustrate potential correlations. We demonstrate how VERB can be used to identify new types of biases such as the *royalty bias* or *nationality bias*.

As an example in Fig. 7, using VERB, one can easily observe that word embeddings capture a clear royalty subspace and resulting bias. For instance, using two-means with seed words "king, queen" (for royal words) and "man, woman" (for common words), a clear royalty subspace becomes quickly apparent. If users also visualize some adjectives as the evaluation set, "obnoxious, considerate, plain, fancy, attentive, important, majestic", as in Fig. 7, they can see a potential bias arising in the captured connotation. Words "obnoxious" and "plain" are more associated with the common direction, while "majestic" is more in the royalty direction.

Note that the y-axis is chosen to show the most variance, and that variation along that direction is not correlated with royalty. Whereas the x-axis is selected to reflect the learned royalty component, and the further left along this coordinate the more common, the further right is more associated with royalty.

Moreover, after removing the royalty concept subspace with Linear Projection (LP), users can observe that the gender concept remains, as shown in Fig. 7 (Right). After another PCA-based reorientation, the words "man" and "king" are to the right, and words "woman" and "queen" to the left. Also, there is still residual gender associations after debiasing. Stereotypical male, chauvinistic traits "important" and "obnoxious" are more on the male side, whereas the stereotypical female subservient traits "considerate" and "attentive" are more associated with the female side.

Another example is shown in Fig. 8 where words encode an axis from North American countries to Middle Eastern countries. The positive vs. negative words are used in an evaluation set, showing that "right" and "good" are more

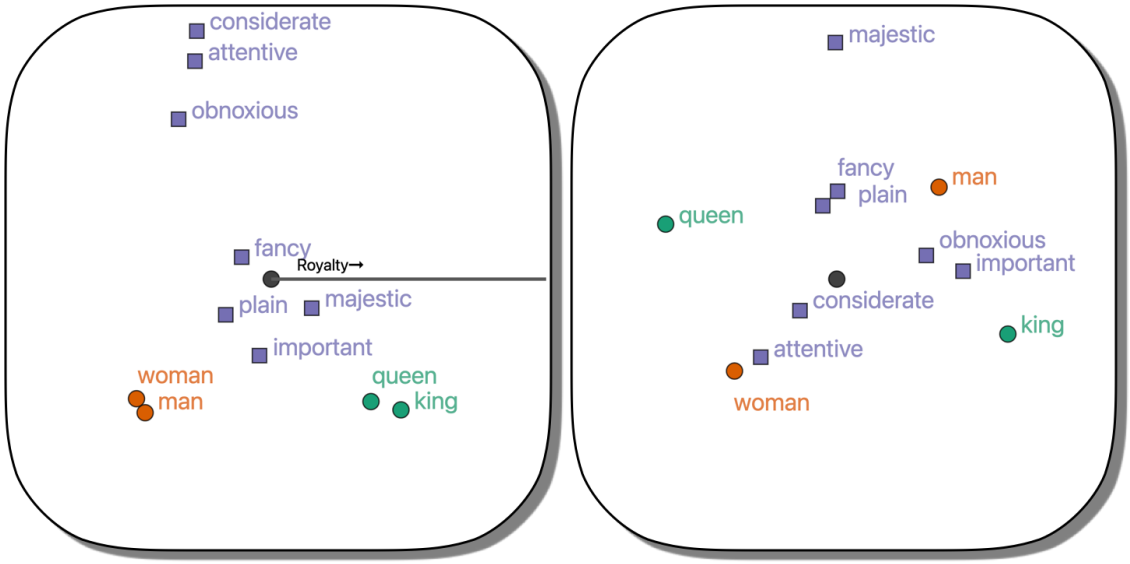


Fig. 7. Left: royalty bias as observed in adjectives, e.g., “majestic” for “queen” and “ king” vs. “obnoxious” for “women” and “man”. Right: the embeddings show residual gender associations after removing the royalty subspace using LP.

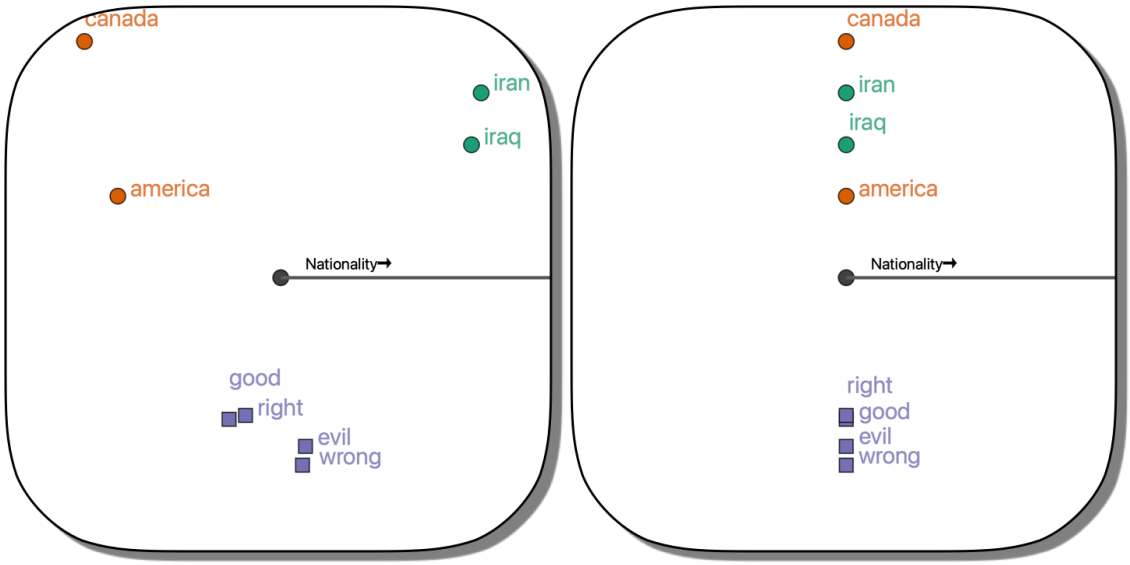


Fig. 8. Left: Nationality bias between North American countries “america”, “canada” vs. Middle East countries “iran”, “iraq” as observed in positive (“good”, “right”) vs. negative (“wrong”, “evil”) words. Right: the removal of this association after applying LP.

associated with North America, whereas “wrong” and “evil” are more associated with the Middle East. Applying LP removes this association.

### 6.2 Explaining Debiasing Methods

VERB was used as part of two recent tutorials at AAAI and KDD. It was also presented as a demo at NeurIPS. In this context, VERB is designed for NLP practitioners who are designing decision-making systems with word embeddings, and also researchers working with the fairness and ethics of ML systems in NLP. It also serves as a visual medium for education, which helps NLP novices to understand and mitigate biases in word embeddings; recently it has been incorporated into a Discover Engineering exhibit slated to represent a new data science major to about 7000 high school students annually in and around Utah.

To explain debiasing methods to these targeted audience, a description of the debiasing methods alone makes it hard for an interested user to decide which one to use, how to use it, and what the limitations are. During this tutorial, participants could easily download VERB, and immediately start interacting with the pre-loaded word vectors embeddings, or creating new examples. They could not only clearly observe the biases and structures presented in these embedded representations, but also compare and contrast their effectiveness and side effects.

Whereas VERB provides utility to run a variety of identification of concept subspaces, and then uses them within debiasing approaches, we highlight a few examples where it is particularly effective in distinguishing variation in the debiasing methods and improving users' understanding of them. In particular, AllenNLP used screenshots from VERB in a preprint of this paper to give an overview of bias mitigation and bias direction methods (see https://guide.allennlp.org/fairness#3).

**Comparing Hard Debiasing vs. Linear Projection.** Bolukbasi et al. [5] introduced the idea of using the Linear Projection (LP) step for debiasing, but wrapped it in a more complex Hard Debiasing (HD) mechanism in an attempt to preserve the structure among definitionally gendered words. This mechanism requires extra word lists and includes a set of paired words, which are equalized. To demonstrate the difference, we use VERB to run HD and LP in Fig. 4 and Fig. 3, respectively, using the same seed sets to define the subspace and evaluation set. HD requires an extra equalize set, and as a result an extra step in the process to equalize those pairs. Hence, it also requires the concept must be the result of some binary notion, thus disallowing concepts like nationality. On the other hand, LP simply projects all words to a subspace that is one dimension lower, including the seed set. Whereas these methods are distinct, it may not be clear all the ways they differ without VERB.

**Understanding OSCaR.** OSCaR [14] is a newer approach to debiasing, and does not rely on projection to remove a subspace. Instead, its operation focuses on a graded rotation (where a different rotation matrix is applied to each point) that, whereas subdifferentiable, requires a complicated case statement to define precisely. With VERB, users are able to, for the first time, dynamically visualize this process under a number of situations. Fig. 6 show snapshots of the process on an example. Particularly in Step 1, users are presented the specific perspective needed to understand the graded rotation, and then it is animated between Steps 1 and 2. Furthermore, users can see how afterwards, both the concepts remain in tact, but they have had their correlation removed.

One observation that quickly became apparent with VERB, but not before, is that OSCaR works more closely to one's intuition (of orthogonalizing subspaces) when the subspaces are defined using PCA (compared to other subspace identification methods). While two-means may do a better job of explicitly capturing the concept subspace for the relationship between two sets (e.g., definitionally male and female words), they do not as explicitly capture a single subspace for the concept as does PCA. Visually, using PCA with OSCaR allows users to easily see the subspace for each concept, where the one for gender can be seen as much less noisy than one for occupations, and how these two subspaces are orthogonal after the operation.

**Residual bias.** A well-known critique of Hard Debiasing [23] is that it leaves residual bias in the embedding, even after the debiasing operation. Whereas such an observation is illustrated mostly quantitatively or abstractly in [23], with VERB, users can easily see the potentially concerning effect. For example, when trying to remove gender bias associated with occupations, HD projects occupation words off the gender-defining subspace. However, for instance, as seen in Fig. 9, the traditional and stereotypical female occupations (e.g., "receptionist" and "homemaker") are still very close to one another, as are stereotypical male professions (e.g., "lawyer" and "engineer"). Such an observation illustrates the concern, since if one knows a homemaker is traditionally female and an engineer is not, then one may infer that so is receptionist, and a lawyer is also not.

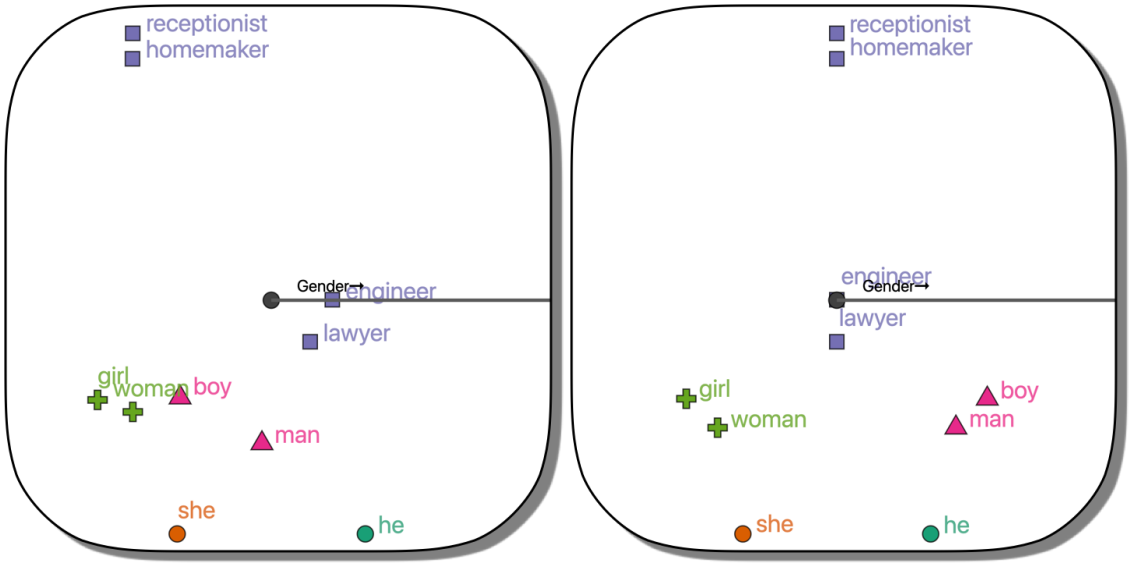


Fig. 9. VERB visualizes residual bias after applying Hard Debiasing.

### 6.3 Interactive Concept Identification

In building VERB, we realize that it provides a new abstraction for understanding existing debiasing methods and creating new ones. In particular, VERB uses a two-step process, subspace identification and embedding transformation, for a debiasing process. Instead of transforming a word vector embedding with a predefined concept subspace, we study how to (a) determine an "optimal" concept subspace and (b) apply transformation *w.r.t.* such a subspace.

As described in Sect. 4, there are several distinct methods to identify concept subspaces (PCA, paired-PCA, two-means, classifier normal). Although these methods are often associated with specific debiasing methods, they are mostly interchangeable. Despite their distinctiveness, the choice of subspace identification method is often ignored or neglected. VERB allows users to easily experiment with these methods, explore their differences and their sensitivity to seed words, and improve the overall effectiveness. Then, going beyond existing mechanisms, VERB provides a novel, optimized concept identification method, improving upon prior methods in quantifiable bias mitigation.

*6.3.1 New Iterative Method of Subspace Identification.* With VERB, we propose a new method of determining a subspace (described by a direction) that captures a specific feature in an embedded representation. Our new method

iteratively improves the subspace direction by optimizing a function. It uses golden-section search (GSS) internally, which given a unimodal function, finds an extremum (minimum or maximum) of the function within a specified interval. It operates by successively narrowing the range of function values within the interval without using the gradient of the function. We use WEAT [8] as the underlying function. That is, given a stereotypical subspace direction $v$, the value $S(v)$ provides the difference in WEAT score *after* applying LP for subspace $v$. The smaller the value, the better $v$ captures the bias subspace.

**Overfitting, testing, and training data.** Before we describe our new subspace identification method, it is important to discuss overfitting, testing data, and training data. According to its formulation, if WEAT is evaluated on sets of words $A$, $B$, $X$, and $Y$, then it may be unfair to train a subspace on the same words it is evaluated on. Otherwise, the trained subspace may not generalize to the vast majority of words not in these sets, i.e., *overfitting*. For example, if we use definitionally gendered words (e.g., "man, woman, boy, girl") in $A$ and $B$ and stereotypical occupation words (e.g, "engineer" and "receptionist") in $X$ and $Y$, then we should consider defining the subspace using words not in those sets. In particular, we use statistically gendered names (i.e., words like "Jack" for male $M$ and "Susan" for female $F$), to define a subspace without using any words that appear in the standard evaluation sets ($A$ and $B$) of definitionally gendered words.

**New iterative subspace identification.** The algorithmic procedure starts with the two-means approach to identifying an initial concept subspace using seed sets $M$ to get mean $m$ and set $F$ to get mean $f$. Initially let $v = m - f$ represent the gender direction. Then, we iteratively improve the WEAT score $S(v)$ by choosing updated points $m$ in the convex combination of $M$ and $f$ in the convex combination of $F$. In each iteration, we fix either $m$ or $f$ and update the other. When $f$ is fixed, we cannot use gradient descent to update $m$, since we do not have access to a gradient of the function $S$. Rather, we consider moving $m$ toward any point $x \in M$, by setting $m$ to its new location $m_x(\alpha) = (1-\alpha)m + \alpha x$ for $\alpha \in [0, 1]$. The parameter $\alpha$ represents the fraction toward $x$ from $m$. We consider each $x \in M$ in a fixed permutation, and determine how best to move $m$ toward $x$, using GSS to optimize $S(m_x(\alpha))$ as a function of $\alpha$. We update $m$ to $m_x(\alpha)$, and then consider the next $x' \in M$ in the permutation, and update $m_x(\alpha)$ to $m_{x'}(\alpha')$ for the best $\alpha'$. After completing this permutation, we fix the new location of $m$, and optimize $f$. These can be alternately optimized. We found that two rounds of optimizing $m$ and $f$ are sufficient.

Whereas the above procedure is automated, VERB is essential in selecting the words used in $M$ and $F$ so we can see how they are correlated with those in $A$ and $B$, respectively. It serves to improve seed word selection, a critical step in a debiasing process.

*6.3.2 Evaluation of Subspace Identification Methods.* We evaluate the effectiveness of our new subspace identification method by computing the WEAT [8] and ECT [16] scores using their respective standard datasets, before and after debiasing with Linear Projection. During the evaluation, we alter the word lists $X$, $Y$ to be stereotypical adjectives so that the evaluation of WEAT is not the same as the one optimized (which used statistically gendered names, and stereotypically gendered occupations to train), that is, we set A = {"strong," "intelligent," "brave," "important"} and B = {"pretty," "beautiful," "shy," "homely"}. In ECT, the aggregated male and female words compare the ranks of distances to a list of occupations. For the classification normal, we debias using Linear Projection exactly once. Recall that a desirable ECT is closer to 1, whereas a desirable WEAT is closer to 0. In addition to our new Iterative Subspace method, we compute the concept direction $v$ using PCA, 2-means, and classification normal, respectively. These are compared against the Baseline of no debiasing. Paired-PCA is not applicable since the input words $M$ and $F$ (statistically gendered names) are not paired. Table 1 shows that for ECT, Iterative Subspace achieves the largest score (nearly the optimal value of 1). Note, if we had trained on $A, B$ instead of $M, F$, then 2-means (which gets the second best score) would

Table 1. Bias Subspace Selection in Word Embeddings.

| Method | ECT | WEAT (adj) | NLI Test |
|---|---|---|---|
| Baseline | 0.773 | 1.587 | 0.297 |
| PCA | 0.905 | 1.17 | 0.346 |
| 2-means | 0.912 | 1.102 | 0.379 |
| Classification (1 step) | 0.872 | 0.951 | 0.383 |
| Iterative Subspace | **0.966** | **0.902** | **0.386** |

get the optimal value of 1. For WEAT, the iterative method (0.902) and the classification normal approach (0.951) both show big improvements over 2-means (1.1) and PCA (1.17). For the more extensive NLI Test [13], we use a large list of gender-occupation bias measuring sentence pairs and record the fraction of sentences classified neutral, a score called Net Neutral [13]. The higher the value (closer to 1), the less the bias (see Sect. 4.3.3; the ideal values for ECT, WEAT and NLI test are 1, 0 and 1, respectively). In this test, we see a similar improvement with the two methods. Iterative subspace identification does the best under all three measures.

### 6.4 VERB for Merchant data

Extensive amounts of transaction data are available to financial organizations. To utilize such data for applications such as recommendation or fraud detection, it is important to understand the characteristics associated with each unique merchant present in the data. Although much information can be obtained by either calculating summary statistics associated with each merchant (e.g., average price for each transaction) or obtaining them directly from the merchant (e.g., merchant categories), a general profile containing additional information can be distilled from the data by creating distributed representations using word embedding algorithms [18, 76, 79]. Such a profile is not based on text associated with these merchants, but rather the sequences in which merchants were visited by customers.

The embedding dataset presented in this section is generated from real-world transactions from a global payment company. It captures payment activities between 70 million merchants and 260 million customers from December 1, 2017 to June 30, 2019 in the United States. The merchant embedding is generated by Word2vec [18, 26], where each merchant is treated as a word and each customer as a document.

With rich information in the merchant embedding, it can be generally applied to many downstream tasks [18, 81]. Here we focus on a subset of the merchant embedding, referred to as the *restaurant embedding*, which is extremely important for recommendation system. By visualizing the merchant embedding dataset through VERB (Fig. 10), we find that the distribution of embeddings are significantly dominated by each restaurants' geographical location over other information such as cuisine type. In other words, geographical location captured by the embedding would interfere with the recommendation system, as a user's "taste" is usually more associated with other information such as cuisine type. It is essential to tease out the undesired subspaces within the embedding space that represents irrelevant information for better recommendation performance.

In Fig. 10 (left), green points are restaurants from the Bay area (BAY), orange points are restaurants from the Los Angeles area (LA), and purple points are the evaluation restaurants. From the visualization, all the LA restaurants are clustered together in the left part of the figure and all the BAY restaurants are clustered in the right part. By using Linear Projection to remove the location subspace and by providing a small number of restaurants in each location as seed sets, similar restaurants are now clustered together, see Fig. 10 (right). For both training dataset and evaluation dataset, we can clearly see seven cuisine types here: Chinese, Ramen, Korean BBQ, Dennys, Donuts, Pizza, Burger/Hotdogs. Therefore,

VERB allows for an easy understanding of restaurant information in the merchant embedding and allows users to modify the embedding to adapt to various downstream tasks, as desired for ⟨R3⟩.

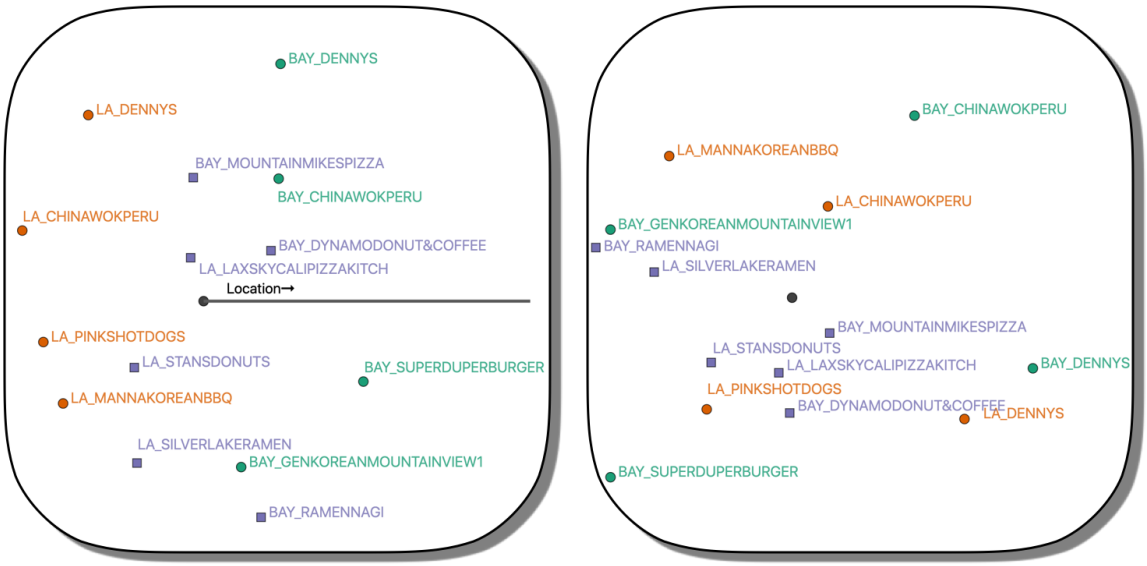


Fig. 10. Removing location information from a restaurant embedding using a Linear Projection debiasing technique. From left to right, visualizing restaurant embedding before and after Linear Projection.

## 7 USER FEEDBACK

We collected both qualitative and quantitative feedback on VERB from the four groups of users with varying backgrounds (described in Sect. 6). The qualitative feedback was conducted through think-aloud discussions with the first three groups of expert users who have thoroughly explored our system. Sect. 6 has described the exploration details and we summarize their feedback in Sect. 7.1. The quantitative feedback was collected through Likert scale questionnaires targeting on the last group of users with varying backgrounds. The analysis result from the questionnaires is reported in Sect. 7.2.

The purpose of reporting the relatively subjective feedback (compared to the more objective case studies in Sec. 6) is to better understand the usability of our system. The feedback from users with varying backgrounds also exposes the usability limitations of our current system, which is useful for its further improvements.

### 7.1 Qualitative Feedback

The qualitative feedback was collected from the first three groups of users after they had comprehensively explored VERB independently on different vector embedding datasets. Specifically, we first introduced VERB to them and explained the functions of different visual components. The users were then given sufficient time (several hours for $W_1$~$W_8$ during the workshop, and a couple of days for $S_1$~$S_2$ and $T_1$~$T_2$). Lastly, feedback was collected from them either in written form and/or through virtual meetings.

In general, all domain users provided positive feedback on VERB in clearly revealing the bias hidden inside embedding data, and intuitively comparing different debiasing algorithms. For example, $W_1$~$W_3$ responded to our feedback questions by valuing the easy accessibility of VERB, and the ease-of-installation was rated between “extremely-easy” and “easy”.

Four of five KDD workshop attendees responded that VERB had changed their views on biases in word representations. As a key takeaway from the KDD tutorial using VERB, $W_4$ commented that they did not know the bias mitigation could be made "that accessible" before the tutorial. $W_5$ was made aware that different types of bias mitigation techniques were available, and they could choose the one that was appropriate for their goals. $W_6$ asked for VERB to allow for supplying their own word embeddings. VERB indeed allows this, as demonstrated in Sect. 6.4. Additional comments included: (a) to use VERB to study biases associated with wealth; (b) to discuss what is lost in the visualization by projecting a 50-dimensional embedding to 2 dimensions; and (c) to investigate further how well the techniques extend to situations where there are (explicitly or not) multiple free variables. Based on a comment from $W_7$, we want to emphasize that gender is not binary, and treating gender as a binary variable in some NLP tasks including debiasing is a simplification (c.f. [15]).

Both $S_1$ and $S_2$ expressed awareness of and worry about the biases in their embedding data. They reaffirmed the importance of the debiasing problem and appreciated the interactivity provided by VERB in easily disclosing the hidden biases. Additionally, we also had thorough discussions with them about other design choices in presenting the clustering results with different dimensionality reduction algorithms. $T_1$ commented that VERB concretized the bias mitigation process in her mind through smooth animations and it demonstrated the process in a more intuitive manner, which significantly helped her understand the merchant embedding data. Compared to the traditional way of manually comparing the embeddings before and after debiasing, VERB is more convenient, interactive, and user-friendly.

There were also several suggested improvements provided by the users. First, $S_2$ discussed the importance of debiasing in language translation tasks to prevent the propagation of bias from one language to the other. He suggested concurrently analyzing multiple sets of embeddings in the same projection space to investigate and relate the biases from individual sets. Adding contexts for biases was another interesting comment, as the biases from one language may not be biases in the other. Second, $S_1$~$S_2$, as well as the workshop attendees, were impressed by VERB integrating so many debiasing algorithms. However, certain technique (e.g., Hard Debiasing) was not very familiar to them. $S_1$ recommended adding short video clips, or links to some algorithm explanations to briefly explain different debiasing techniques. Lastly, some direct side-by-side comparisons for different debiasing algorithms were also recommended by the users (without considering the space usage). The feedback provided promising further directions for us to explore.

### 7.2 Quantitative Feedback

Apart from the qualitative feedback collected through guided exploration and think-aloud discussions in Sect. 7.1, we also wanted to solicit users' quantitative opinions on the usability of VERB. While there were a number of visualization evaluation approaches [19], according to the visualization literature [29, 34], Likert-scale questionnaires [65] are often the ideal choices to collect users' subjective opinions, especially when evaluating the user experience on a visualization tool. The collective behaviors of individual responses to the questionnaires would constitute quantitative feedback from the users.

In detail, we conducted multiple VERB exploration sessions with the last group of users, i.e., 20+ research scientists and Ph.D. students from a research institute. The users participated the sessions voluntarily and the sessions were carried out in a setting similar to that described in Sect. 7.1, i.e., we explained different debiasing algorithms and how VERB works, and provided detailed instructions on its usage. The participants were asked to explore the system for a sufficient amount of time and fill out a survey with Likert scale questions (attached in the supplementary material).

From the 20+ participants, 20 valid responses were collected. A summary of the participants' background and responses is shown in Fig. 11. Most of the participants had ML and NLP backgrounds and half of the participants self-identified

as “knowledgeable” in bias and model fairness analysis (Fig. 11, left). The Likert scale questions solicited the users’ agreement level on the following questions:

- Q1: VERB is superior to existing solutions that I used before in easily identifying biases in embedding data (Bias Identification).
- Q2: The debiasing process presented by VERB is intuitive and helps me understand and compare different debiasing algorithms (Bias Understanding and Comparison).
- Q3: I feel engaged in the bias exploration process and am confident in verifying the quality of the debiased embeddings (Bias Exploration, Verification).
- Q4: VERB is easily accessible, and the interactions are user-friendly (Accessibility and Usability).

These questions are designed to reflect certain aspects of the user experience that we care about the most, covering the usability, user-engagement, and accessibility of VERB. However, these preliminary questions may inevitably provide users with hints about our expected responses and more thoughts and iterations are needed to further refine them in the future. For example, users’ confidence level in verifying the quality of debiased embedding (Q3) may involve multiple factors, e.g., their knowledge in this area, their familiarity with the dataset, and the performance of the system. Bearing these limitations in mind, we still would like to obtain some initial impressions on how users feel about VERB and reflect the users’ engagement level over the explorations. When conducting the study, we tried our best to mitigate the biases of the questionnaires by asking users to voluntarily participate in the exploration sessions and encouraging them to answer the questions more objectively. The result from this preliminary user study is very encouraging. The known limitations of the current study would lead to improved user studies in the future, such as carefully-designed questions to minimize influence by expectations, and comprehensive metrics to more directly and objectively reflect the system performance.

From the survey, we found that the users agreed with our stated claim that VERB can help to identify, understand, and compare biases in word embeddings (Fig. 11, right); there was no disagreement. They also provided positive comments on the intuitive user interactions and exploration processes. There were some constructive suggestions. As the users had varying levels of knowledge on debiasing algorithms, some users suggested the inclusion of more explanations or links to external resources of the algorithms. This was consistent with our findings from Sect. 7.1, and we will integrate these suggestions into the next version of VERB.

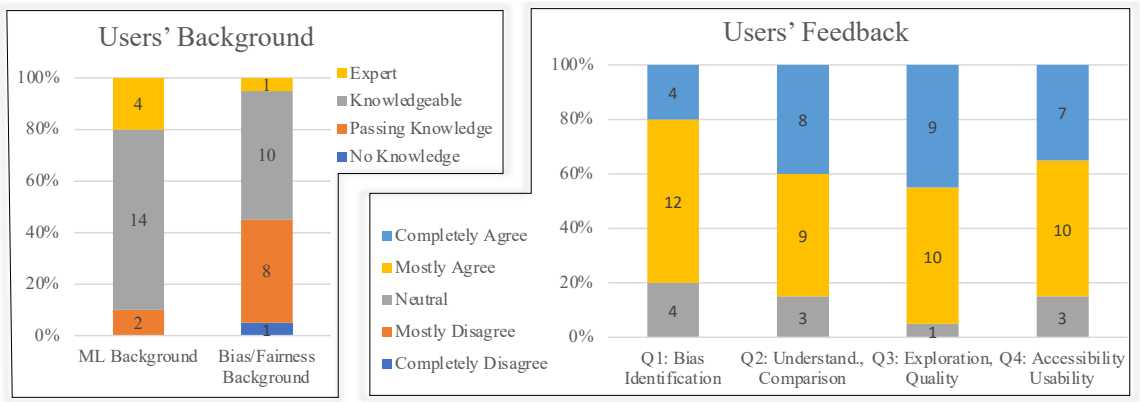


Fig. 11. Users’ background and responses to our questions during the quantitative evaluation.

## 8 DISCUSSION AND CONCLUSIONS

This paper presents VERB, a new tool for visualizing, interacting with, and teaching embedded representations of data. It is especially useful at allowing users to interact and observe the effects of debiasing word vector embeddings – an essential component of most NLP tasks, and critical for ensuring fairness.

VERB has been deployed as a central element of recent tutorials at AAAI and KDD, aiding the research in understanding embeddings within both academia and industry. It was also presented as a NeurIPS demo. The tool has been a part of several seminar talks directed toward various audiences ranging from students to senior data scientists, e.g., the Utah Data Science Seminar and the Visualization Seminar. VERB has been utilized in a female-focused summer camp, the Hi-GEAR (Girls Engineering Abilities Realized) summer camp 2021, which exposes high-schoolers (currently in 9th-12th grade) to a variety of engineering and computer science careers with hands-on experiential learning and collaborative team projects. Most recently, VERB has been integrated into a "Discover Engineering" program that was presented to around 7,000 high school students interested in engineering in and around Utah, as an entry way into a new data science major.

VERB is distinct from previous visualization tools for high-dimensional vectorized data in that it is used to modify and improve the vectors, before they are used in downstream tasks, not just explore or inspect them. As demonstrated in Sect. 6, the VERB visual tool is useful in other ways. It allows us to find new types of bias. It identifies the importance of subspace determination in addition to the debiasing mechanisms – leading to an improved method. Furthermore, it has been demonstrated to extend to other domains such as merchant data from financial transactions. Due to its generality and modularity, it is possible to apply VERB to other domains that utilize vectorized data representations.

A key challenge VERB addresses is how to show high-dimensional embedded represention informatively when only a 2-dimensional view is visually available. It relies on displaying this vectorized data as projected views, via *linear* projections. This linearity is essential since the debiasing and modification relies on identification and movement of data along linear subspaces. Nonlinear methods such as t-SNE or ISOMAP would distort these linear objects. Moreover, these identified concept subspaces are essential to defining these views. These identified subspaces define the $x$- and possibly $y$-axis of the view, where users can clearly see the amount of contribution individual words have along that subspace, without concerning the possible distortions introduced by a skewed projection.

Our proposed system has limitations in its current iteration, a few of which are identified below. First, VERB has been evaluated through feedback from guided explorations and interviews with domain experts. In the future, we will consider statistical user studies focusing on various use cases to identify and improve upon limitations of the tool. Second, due to fundamental differences of the debiasing algorithms, VERB does not allow a simultaneous (side-by-side) visual comparison of multiple debiasing algorithms' outputs, although it does report the single most common WEAT score for each algorithm before and after debiasing. Instead, VERB allows users to export the embeddings for their own analysis pipelines. Due to the diversity of potential analysis one might want to perform, it would be hard to incorporate all potential downstream analysis into VERB. Third, the seed sets for the debiasing methods have to be manually entered. It may be helpful to provide suggestions for expanded seed sets to users based on their presently entered words. It is, however, prone to introducing additional bias and will warrant careful design of such a feature. Fourth, it would be useful to help users evaluate parameter choices semiautomatically through the VERB system for cases when the use of a particular debiasing algorithm and subspace identification is unclear, which is left for future work. VERB relies on the human-in-the-loop to identify biases for exploration. Automatic identification of (unknown) biases is nontrivial and remains elusive for the current system. Finally, whereas the majority of existing methods assume bias to be binary for simplicity, bias can be

multifaceted in nature. Adapting our system to handle multiple facets of bias as more sophisticated debiasing methods are introduced would be another future direction.

By revisiting our design requirements (R1-R3), the lessons learned during the design and evaluation process of VERB may offer valuable insights that can inform the design of future tools for ML interpretability. The first lesson is based on the principle of "seeing is believing". Users need to be shown that various biases indeed exist through a hands-on experience in order to appreciate the efforts behind bias mitigation algorithms. Following the visualization pipeline for high-dimensional data [40], the second lesson is to use intuitive operations during visual mapping. Our visualization relies on displaying the data via linear projections to preserve users' geometric intuitions. The third lesson is to decompose a well-encapsulated process into modular, simple components for individual demonstration and better interpretation. The final lesson is to create a tool that is modular and easily generalizable. VERB can be used to explore any high-dimensional data embeddings beyond those that arise from NLP tasks, as shown in Sect. 6.4 for financial data. Furthermore, it may be extended to support languages beyond English or even multilingual models, as long as embedding files are provided and seed sets are modified accordingly.

With the rise of contextualized word embeddings, it remains useful to study non-contextual embeddings using tools such as VERB. Some applications still use non-contextual word embeddings (over contextual ones) for their simplicity, interpretability, and efficiency. Many domains (such as analyzing merchants) use embeddings as a primary data representation, and have yet to identify meaningful ways to extend to contextual embeddings. Many of the debiasing techniques for contextual embeddings are direct extensions of those incorporated into VERB. It is even more difficult to visualize and interpret contextual embeddings as compared to non-contextual ones. Therefore as an education tool, even for understanding how to work with contextual embeddings, VERB on non-contextual embeddings is still important to build geometric intuitions. Furthermore, the first layer of any contextualized encoders such as BERT [17] and RoBERTa [47], as well as the last layer of a decoder such as GPT-2/3 [6] or the output of models in the T5 family, are essentially token or word-piece embedding matrices. VERB could provide an interactive way to investigate biases in these layers. Therefore, the design of VERB, and the lessons learned from its design and evaluation, can be extended to the the now commonly used contextualized embeddings such as BERT, RoBERTa, and their derivatives. The full exploration of this is left for future work.

Overall, VERB is a simple and easy-to-use interface for understanding and acting on a wide variety of vectorized representations. It provides a powerful way to demystify, interact with, and apply debiasing to these representations, toward interpretable and fair ML systems that operate on these representations.

## ACKNOWLEDGMENTS

This project was partially supported by the Utah Board of Higher Education's Deep Technology Initiative under the project "Bringing Fairness in AI to the Forefront of Education". It was also partially supported by a grant from VISA Research, and NSF grants 1514520, 1564287, 1350888, 2115677, 1816149, 1822877, 2134223, 1661375, 2007398, 1822877, 2007398, and 2205418.

## REFERENCES

[1] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. 2016. TensorFlow: A System for Large-Scale Machine Learning. In *Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI)*.

[2] Saleema Amershi, Max Chickering, Steven M Drucker, Bongshin Lee, Patrice Simard, and Jina Suh. 2015. ModelTracker: Redesigning Performance Analysis Tools for Machine Learning. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. 337–346.

[3] Alejandro Barredo Arrieta, Natalia Díaz-Rodríguez, Javier Del Ser, Adrien Bennetot, Siham Tabik, Alberto Barbado, Salvador García, Sergio Gil-López, Daniel Molina, Richard Benjamins, et al. 2020. Explainable Artificial Intelligence (XAI): Concepts, Taxonomies, Opportunities and Challenges toward Responsible AI. *Information Fusion* 58 (2020), 82–115.

[4] Su Lin Blodgett, Solon Barocas, Hal Daumé III, and Hanna Wallach. 2020. Language (Technology) is Power: A Critical Survey of "Bias" in NLP. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 5454–5476.

[5] Tolga Bolukbasi, Kai-Wei Chang, James Zou, Venkatesh Saligrama, and Adam Kalai. 2016. Man is to Computer Programmer as Woman is to Homemaker? Debiasing Word Embeddings. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*. 4356–4364.

[6] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language Models are Few-Shot Learners. In *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (Eds.), Vol. 33. Curran Associates, Inc., 1877–1901.

[7] Ángel Alexander Cabrera, Will Epperson, Fred Hohman, Minsuk Kahng, Jamie Morgenstern, and Duen Horng Chau. 2019. FAIRVIS: Visual Analytics for Discovering Intersectional Bias in Machine Learning. In *IEEE Conference on Visual Analytics Science and Technology (VAST)*. 46–56.

[8] Aylin Caliskan, Joanna J. Bryson, and Arvind Narayanan. 2017. Semantics Derived Automatically From Language Corpora Contain Human-Like Biases. *Science* 356, 6334 (2017), 183–186.

[9] Changjian Chen, Jun Yuan, Yafeng Lu, Yang Liu, Hang Su, Songtao Yuan, and Shixia Liu. 2021. OoDAnalyzer: Interactive Analysis of Out-Of-Distribution Samples. *IEEE Transactions on Visualization and Computer Graphics* 27, 1 (2021), 3335–3349.

[10] Jaegul Choo and Shixia Liu. 2018. Visual Analytics for Explainable Deep Learning. *IEEE Transactions on Visualization and Computer Graphics* 38, 4 (2018), 84–92.

[11] Jason Chuang, Christopher D Manning, and Jeffrey Heer. 2012. Termite: Visualization Techniques for Assessing Textual Topic Models. In *Proceedings of the International Working Conference on Advanced Visual Interfaces*. 74–77.

[12] Sunipa Dev, Saffia Hassan, and Jeff M. Phillips. 2019. Closed Form Word Embedding Alignment. In *International Conference on Data Mining (ICDM)*. 130–139.

[13] Sunipa Dev, Tao Li, Jeff Phillips, and Vivek Srikumar. 2020. On Measuring and Mitigating Biased Inferences of Word Embeddings. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 7659–7666.

[14] Sunipa Dev, Tao Li, Jeff M Phillips, and Vivek Srikumar. 2021. OSCaR: Orthogonal Subspace Correction and Rectification of Biases in Word Embeddings. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 5034–5050.

[15] Sunipa Dev, Masoud Monajatipoor, Anaelia Ovalle, Arjun Subramonian, Jeff Phillips, and Kai-Wei Chang. 2021. Harms of Gender Exclusivity and Challenges in Non-Binary Representation in Language Technologies. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 1968–1994.

[16] Sunipa Dev and Jeff M. Phillips. 2019. Attenuating Bias in Word vectors. In *International Conference on Artificial Intelligence and Statistics (Proceedings of Machine Learning Research)*. PMLR, 879–887.

[17] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding. *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (2019), 4171–4186.

[18] Min Du, Robert Christensen, Wei Zhang, and Feifei Li. 2019. Pcard: Personalized Restaurants Recommendationfrom Card Payment Transaction Records. In *The World Wide Web Conference*. 2687–2693.

[19] Niklas Elmqvist and Ji Soo Yi. 2015. Patterns for Visualization Evaluation. *Information Visualization* 14, 3 (2015), 250–269.

[20] Niklas Friedrich, Anne Lauscher, Simone Paolo Ponzetto, and Goran Glavaš. 2021. DebIE: A Platform for Implicit and Explicit Debiasing of Word Embedding Spaces. *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations* (2021), 91–98.

[21] Bhavya Ghai, Md Naimul Hoque, and Klaus Mueller. 2021. WordBias: An Interactive Visual Tool for Discovering Intersectional Biases Encoded in Word Embeddings. In *Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems*. 1–7.

[22] Aindrila Ghosh, Mona Nashaat, James Miller, and Shaikh Quader. 2020. VisExPreS: A Visual Interactive Toolkit for User-driven Evaluations of Embeddings. *IEEE Transactions on Visualization and Computer Graphics* 28, 7 (2020), 2791–2807.

[23] 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 (NAACL-HLT)*. 609–614.

[24] Google. 2020. What-If Tool. https://pair-code.github.io/what-if-tool/.

[25] Google-PAIR. 2017. Facets: Visualizations for Machine Learing Datasets. https://pair-code.github.io/facets/.

[26] Florian Heimerl and Michael Gleicher. 2018. Interactive Analysis of Word Vector Embeddings. *Computer Graphics Forum* 37, 3 (2018), 253–265.

[27] Florian Heimerl, Christoph Kralj, Torsten Moller, and Michael Gleicher. 2020. embComp: Visual Interactive Comparison of Vector Embeddings. *IEEE Transactions on Visualization and Computer Graphics* 28, 8 (2020), 2953–2969.

[28] Andreas Hinterreiter, Peter Ruch, Holger Stitz, Martin Ennemoser, Jurgen Bernard, Hendrik Strobelt, and Marc Streit. 2020. ConfusionFlow: A Model-Agnostic Visualization for Temporal Analysis of Classifier Confusion. *IEEE Transactions on Visualization and Computer Graphics* 28, 2 (2020), 1222–1236.

[29] Tobias Isenberg, Petra Isenberg, Jian Chen, Michael Sedlmair, and Torsten Möller. 2013. A Systematic Review on the Practice of Evaluating Visualization. *IEEE Transactions on Visualization and Computer Graphics* 19, 12 (2013), 2818–2827.

[30] Minsuk Kahng, Dezhi Fang, and Duen Horng (Polo) Chau. 2016. Visual Exploration of Machine Learning Results Using Data Cube Analysis. In *Proceedings of the Workshop on Human-In-the-Loop Data Analytics*. 1–6.

[31] Hannah Kim, Jaegul Choo, Haesun Park, and Alex Endert. 2015. InterAxis: Steering Scatterplot Axesvia Observation-Level Interaction. *IEEE Transactions on Visualization and Computer Graphics* 22, 1 (2015), 131–140.

[32] Josua Krause, Adam Perer, and Enrico Bertini. 2014. INFUSE: Interactive Feature Selection for Predictive Modeling of High Dimensional Data. *IEEE Transactions on Visualization and Computer Graphics* 20, 12 (2014), 1614–1623.

[33] Matt J. Kusner, Joshua Loftus, Chris Russell, and Ricardo Silva. 2017. Counterfactual Fairness. *Advances in Neural Information Processing Systems* (2017), 4066–4076.

[34] Heidi Lam, Enrico Bertini, Petra Isenberg, Catherine Plaisant, and Sheelagh Carpendale. 2011. Empirical Studies in Information Visualization: Seven Scenarios. *IEEE Transactions on Visualization and Computer Graphics* 18, 9 (2011), 1520–1536.

[35] Quan Li, Kristanto Sean Njotoprawiro, Hammad Haleem, Qiaoan Chen, Chris Yi, and Xiaojuan Ma. 2018. EmbeddingVis: A Visual Analytics Approach to Comparative Network Embedding Inspection. In *IEEE Conference on Visual Analytics Science and Technology (VAST)*. 48–59.

[36] Mengchen Liu, Jiaxin Shi, Zhen Li, Chongxuan Li, Jun Zhu, and Shixia Liu. 2016. Towards Better Analysis of Deep Convolutional Neural Networks. *IEEE Transactions on Visualization and Computer Graphics* 23, 1 (2016), 91–100.

[37] Shusen Liu, Peer-Timo Bremer, Jayaraman J. Thiagarajan, Vivek Srikumar, Bei Wang, Yarden Livnat, and Valerio Pascucci. 2018. Visual Exploration of Semantic Relationships in Neural Word Embeddings. *IEEE Transactions on Visualization and Computer Graphics* 24, 1 (2018), 553–562.

[38] Shusen Liu, Tao Li, Zhimin Liu, Vivek Srikumar, Valerio Pascucci, and Peer-Timo Bremer. 2018. Visual Interrogation of Attention-Based Models for Natural Language Inference and Machine Comprehension. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP): System Demonstrations*. 36–41.

[39] Shusen Liu, Zhimin Li, Tao Li, Vivek Srikumar, Valerio Pascucci, and Peer-Timo Bremer. 2019. NLIZE: A Perturbation-Driven Visual Interrogation Tool for Analyzing and Interpreting Natural Language Inference Models. *IEEE Transactions on Visualization and Computer Graphics* 25, 1 (2019), 651–660.

[40] Shusen Liu, Dan Maljovec, Bei Wang, Peer-Timo Bremer, and Valerio Pascucci. 2017. Visualizing High-Dimensional Data: Advances in the Past Decade. *IEEE Transactions on Visualization and Computer Graphics* 23, 3 (2017), 1249–1268.

[41] Shixia Liu, Xiting Wang, Mengchen Liu, and Jun Zhu. 2017. Towards Better Analysis of Machine Learning Models: A Visual Analytics Perspective. *Visual Informatics* 1, 1 (2017), 48–56.

[42] Shixia Liu, Yingcai Wu, Enxun Wei, Mengchen Liu, and Yang Liu. 2013. StoryFlow: Tracking the Evolution of Stories. *IEEE Transactions on Visualization and Computer Graphics* 19, 12 (2013), 2436–2445.

[43] Shixia Liu, Jiannan Xiao, Junlin Liu, Xiting Wang, Jing Wu, and Jun Zhu. 2017. Visual Diagnosis of Tree Boosting Methods. *IEEE Transactions on Visualization and Computer Graphics* 24, 1 (2017), 163–173.

[44] Shixia Liu, Jialun Yin, Xiting Wang, Weiwei Cui, Kelei Cao, and Jian Pei. 2016. Online Visual Analytics of Text Streams. *IEEE Transactions on Visualization and Computer Graphics* 22, 11 (2016), 2451–2466.

[45] Xiao Liu and Junpeng Wang. 2020. LatentVis: Investigating and Comparing Variational Auto-Encoders via Their Latent Space. In *Proceedings of the CIKM Workshop on Advances in Interpretable Machine Learning and Artificial Intelligence (AIMLAI)*.

[46] Yang Liu, Eunice Jun, Qisheng Li, and Jeffrey Heer. 2019. Latent Space Cartography: Visual Analysis of Vector Space Embeddings. *Computer Graphics Forum* 38, 3 (2019), 67–78.

[47] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv preprint arXiv:1907.11692* (2019).

[48] Thorsten May, Andreas Bannach, James Davey, Tobias Ruppert, and Jörn Kohlhammer. 2011. Guiding Feature Subset Selection With an Interactive Visualization. In *IEEE Conference on Visual Analytics Science and Technology (VAST)*. 111–120.

[49] Leland McInnes, John Healy, and James Melville. 2018. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. arXiv preprint arXiv:1802.03426.

[50] Leland McInnes, John Healy, Nathaniel Saul, and Lukas Großberger. 2018. UMAP: Uniform Manifold Approximation and Projection. *Journal of Open Source Software* 3, 29 (2018), 861.

[51] Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013. Distributed Representations of Words and Phrases and Their Compositionality. *Advances in Neural Information Processing Systems 26* (2013), 3111–3119.

[52] Ankur Parikh, Oscar Täckström, Dipanjan Das, and Jakob Uszkoreit. 2016. A Decomposable Attention Model for Natural Language Inference. *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)* (2016).

[53] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12 (2011), 2825–2830.

[54] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global Vectors for Word Representation. *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)* (2014), 1532–1543.

[55] Nicola Pezzotti, Thomas Höllt, Jan Van Gemert, Boudewijn PF Lelieveldt, Elmar Eisemann, and Anna Vilanova. 2017. DeepEyes: Progressive Visual Analytics for Designing Deep Neural Networks. *IEEE Transactions on Visualization and Computer Graphics* 24, 1 (2017), 98–108.

[56] James Powell, Kari Sentz, and Martin Klein. 2021. Human-in-the-Loop Refinement of Word Embeddings. arXiv preprint arXiv:2110.02884.

[57] Archit Rathore, Nithin Chalapathi, Sourabh Palande, and Bei Wang. 2021. TopoAct: Exploring the Shape of Activations in Deep Learning. *Computer Graphics Forum* 40, 1 (2021), 382–397.

[58] Paulo E Rauber, Samuel G Fadel, Alexandre X Falcao, and Alexandru C Telea. 2016. Visualizing the Hidden Activity of Artificial Neural Networks. *IEEE Transactions on Visualization and Computer Graphics* 23, 1 (2016), 101–110.

[59] Shauli Ravfogel, Yanai Elazar, Hila Gonen, Michael Twiton, and Yoav Goldberg. 2020. Null It Out: Guarding Protected Attributes by Iterative Nullspace Projection. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics* (2020), 7237–7256.

[60] Donghao Ren, Saleema Amershi, Bongshin Lee, Jina Suh, and Jason D Williams. 2016. Squares: Supporting Interactive Performance Analysis for Multiclass Classifiers. *IEEE Transactions on Visualization and Computer Graphics* 23, 1 (2016), 61–70.

[61] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. Why Should I Trust You? Explaining the Predictions of Any Classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 1135–1144.

[62] Daniel Smilkov, Nikhil Thorat, Charles Nicholson, Emily Reif, Fernanda B Viégas, and Martin Wattenberg. 2016. Embedding Projector: Interactive Visualization and Interpretation of Embeddings. arXiv preprint arXiv:1611.05469.

[63] Alison Smith, Jason Chuang, Yuening Hu, Jordan Boyd-Graber, and Leah Findlater. 2014. Concurrent Visualization of Relationships between Words and Topics in Topic Models. In *Proceedings of the Workshop on Interactive Language Learning, Visualization, and Interfaces*. 79–82.

[64] Alison Smith, Timothy Hawes, and Meredith Myers. 2014. Hiérarchie: Interactive Visualization for Hierarchical Topic Models. In *Proceedings of the Workshop on Interactive Language Learning, Visualization, and Interfaces*. 71–78.

[65] Laura South, David Saffo, Olga Vitek, Cody Dunne, and Michelle A Borkin. 2022. Effective Use of Likert Scales in Visualization Evaluations: A Systematic Review. In *Computer Graphics Forum*, Vol. 41. 43–55.

[66] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. 2015. Striving for Simplicity: The All Convolutional Net. In *3rd International Conference on Learning Representations, Workshop Track Proceedings*.

[67] Tony Sun, Andrew Gaut, Shirlyn Tang, Yuxin Huang, Mai ElSherief, Jieyu Zhao, Diba Mirza, Elizabeth M. Belding, Kai-Wei Chang, and William Yang Wang. 2019. Mitigating Gender Bias in Natural Language Processing: Literature Review. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Florence, Italy, 1630–1640.

[68] Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing Data Using t-SNE. *Journal of Machine Learning Research* 9, 2579-2605 (2008), 85.

[69] Emily Wall, Leslie Blaha, Celeste Paul, Kristin Cook, and Alex Endert. 2018. Four Perspectives on Human Bias in Visual Analytics. In *Cognitive Biases in Visualizations*, Geoffrey Ellis (Ed.). Springer Nature Switzerland, Springer, Cham, 29–42.

[70] Emily Wall, John Stasko, and Alex Endert. 2019. Toward a Design Space for Mitigating Cognitive Bias in Vis. In *IEEE Visualization Conference (VIS)*. 111–115.

[71] Changbo Wang, Zhao Xiao, Yuhua Liu, Yanru Xu, Aoying Zhou, and Kang Zhang. 2013. SentiView: Sentiment Analysis and Visualization for Internet Popular Topics. *IEEE Transactions on Human-Machine Systems* 43, 6 (2013), 620–630.

[72] Evan Wang. 2013. *A D3 Plug-in for Automatic Label Placement Using Simulated Annealing*. Technical Report CS294-10-fa13 (coursenotes). University of California, Berkeley.

[73] Junpeng Wang, Liang Gou, Hao Yang, and Han-Wei Shen. 2018. GANViz: A Visual Analytics Approach to Understand the Adversarial Game. *IEEE Transactions on Visualization and Computer Graphics* 24, 6 (2018), 1905–1917.

[74] Junpeng Wang, Wei Zhang, and Hao Yang. 2020. SCANViz: Interpreting the Symbol-Concept Association Captured by Deep Neural Networks through Visual Analytics. In *IEEE Pacific Visualization Symposium (PacificVis)*. 51–60.

[75] 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*. 5443–5453.

[76] Yuwei Wang, Yan Zheng, Yanqing Peng, Michael Yeh, Zhongfang Zhuang, Das Mahashweta, Bendre Mangesh, Feifei Li, Wei Zhang, and Jeff M. Phillips. 2021. Constrained Non-Affine Alignment of Embeddings. In *IEEE International Conference on Data Mining (ICDM)*. 1403–1408.

[77] Kanit Wongsuphasawat, Daniel Smilkov, James Wexler, Jimbo Wilson, Dandelion Mane, Doug Fritz, Dilip Krishnan, Fernanda B Viégas, and Martin Wattenberg. 2017. Visualizing Dataflow Graphs of Deep Learning Models in TensorFlow. *IEEE Transactions on Visualization and Computer Graphics* 24, 1 (2017), 1–12.

[78] Weikai Yang, Zhen Li, Mengchen Liu, Yafeng Lu, Kelei Cao, Ross Maciejewski, and Shixia Liu. 2020. Diagnosing Concept Drift with Visual Analytics. In *IEEE Conference on Visual Analytics Science and Technology (VAST)*. 12–23.

[79] Chin-Chia Michael Yeh, Dhruv Gelda, Zhongfang Zhuang, Yan Zheng, Liang Gou, and Wei Zhang. 2020. Towards a Flexible Embedding Learning Framework. In *International Conference on Data Mining Workshops (ICDMW)*, Vol. 1. 605–612.

[80] Jun Yuan, Changjian Chen, Weikai Yang, Mengchen Liu, Jiazhi Xia, and Shixia Liu. 2021. A Survey of Visual Analytics Techniques for Machine Learning. *Computational Visual Media* 7, 1 (2021), 3–36.

[81] Wei Zhang, Liang Wang, Robert Christensen, Yan Zheng, Liang Gou, and Hao Yang. U.S. Patent 20200314101, Oct. 2020. Transaction Sequence Processing With Embedded Real-Time Decision Feedback. https://www.freepatentsonline.com/y2020/0314101.html.

[82] Jieyu Zhao, Yichao Zhou, Zeyu Li, Wei Wang, and Kai-Wei Chang. 2018. Learning Gender-Neutral Word Embeddings. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. 4847–4853.