Fairness in Embedding-Based Machine Learning Models

Adam Ge
Mentor: Mayuri Sridhar

MIT PRIMES 2025

Abstract

Fairness in machine learning has become a critical concern, particularly for decision making systems that rely on learned representations and are trained on data containing historical and societal biases. In this work, we study fairness in embedding-based models from two complementary perspectives. First, we examine gender bias in text embeddings produced by pretrained language models and propose a baseline method based on sparse autoencoders to disentangle a gender-related feature and mitigate bias at the embedding level. While effective for natural language data, this approach relies on carefully constructed contrasting examples and is difficult to extend to other data modalities.

To address this limitation, we propose *IterativeSifting*, a general framework for improving fairness in embedding-based decision making models. IterativeSifting iteratively identifies and removes latent features and proxy information associated with one or more sensitive attributes, including their intersections, while preserving task-relevant information for accurate prediction. The method is model-agnostic and applicable to a wide range of data types, including tabular and graph-structured data.

We evaluate IterativeSifting on standard fairness benchmarks, including the Adult Census Income and ACSIncome datasets, using gender and race as sensitive attributes. Experimental results show that IterativeSifting substantially reduces sensitive attribute information in learned embeddings and significantly improves intersectional fairness, as measured by mutual information and maximum equalized odds difference, while maintaining competitive predictive performance. These results demonstrate the effectiveness of IterativeSifting as a practical approach for mitigating bias in embedding-based decision making systems.

1 Introduction

Machine learning models are increasingly used to support high-stakes decision making tasks, such as hiring, admissions, and content moderation [1]. In these settings, models are often trained on large datasets that reflect historical and

societal biases, which may lead to unfair or discriminatory outcomes for individuals associated with sensitive attributes such as gender or race [2]. A central challenge in addressing these issues is that modern models typically operate on learned latent representations, or embeddings, in which task-relevant information and sensitive attribute information are deeply entangled [12]. Understanding how bias manifests in embeddings, and how it can be measured and mitigated without substantially degrading predictive performance, is therefore a key problem in the study of fairness in machine learning.

Related challenges arise in settings where it is desirable to remove or suppress specific information from a trained model without retraining it from scratch. In prior work, I studied the removal of relational information, such as edges, from graph neural network models, where structural dependencies in the data complicate the isolation of individual relationships. Although the focus of that work differs from fairness, a similar phenomenon appears in embedding-based models: information of interest is not explicitly stored, but is implicitly encoded in learned representations. In the context of fairness, sensitive attribute information may be diffusely embedded across latent features, making it difficult to reduce bias without affecting task-relevant performance. This report builds on these insights to study how bias manifests in embeddings and how it can be measured and mitigated in both language models and decision making systems.

Pretrained language models are now widely used in modern machine learning systems. We first investigate bias and fairness in the text embeddings produced by these models. As a baseline, we propose a method based on *sparse autoencoders* (SAEs) [3] to disentangle a sensitive attribute, such as *gender*, and to reveal systematic biases in word embeddings. We further introduce two strategies for mitigating such biases at the embedding level. However, this approach relies on carefully curated contrasting examples, which may be difficult or infeasible to obtain for many data modalities.

To address these limitations, we propose *IterativeSifting*, a general framework for improving fairness in embedding-based decision making models. IterativeSifting operates directly on learned representations and does not require carefully constructed contrasting examples. Instead, it iteratively identifies and removes latent features and proxy information associated with one or more sensitive attributes, such as gender and race, while preserving the information necessary for accurate task prediction. The method is model-agnostic and can be applied to a wide range of data modalities, including tabular data, graphs, and text, making it suitable for real-world decision making scenarios where sensitive attributes and their proxies are deeply entangled in the data.

We evaluate IterativeSifting on standard fairness benchmarks, including the Adult Census Income dataset [22] and the ACSIncome dataset [23], considering both gender and race as sensitive attributes. Experimental results show that IterativeSifting substantially reduces sensitive attribute information in the learned embeddings, as measured by mutual information, and significantly improves intersectional fairness at the decision level, as reflected by reduced maximum equalized odds differences [7] across demographic groups. Importantly, these fairness gains are achieved while maintaining competitive predictive accu-

racy on the target task, demonstrating that IterativeSifting effectively balances fairness and utility in embedding-based decision making models.

2 Related Work

Fairness in machine learning has been widely studied in the context of algorithmic decision making systems that affect individuals, such as hiring, lending, and criminal justice. A foundational line of work formalizes different notions of fairness at the level of model outputs, including demographic parity, equal opportunity, and equalized odds [7, 8]. Among these, equalized odds is particularly popular because it controls disparities in both true positive and false positive rates across sensitive groups. Recent work has further emphasized the importance of evaluating fairness over intersectional groups formed by multiple sensitive attributes, as fairness guarantees with respect to individual attributes may fail for their intersections [9, 10].

Fair representation learning. A major line of research focuses on learning fair or invariant representations that remove sensitive attribute information while preserving task-relevant features. One influential approach is adversarial debiasing, where an encoder is trained to be predictive of the target task while simultaneously preventing an adversarial discriminator from predicting sensitive attributes [11, 12, 13]. Such adversarial methods have been applied across various data modalities, including tabular data, images, and graphs [14]. However, adversarial training can be unstable and often requires careful tuning to balance fairness and utility.

Other representation-level approaches explicitly regularize the dependence between learned representations and sensitive attributes, for example by minimizing mutual information or enforcing independence constraints [15, 16]. These methods are closely related to information bottleneck principles and offer a more direct way to quantify sensitive attribute leakage. Nevertheless, many existing approaches focus on a single sensitive attribute and do not naturally extend to intersectional fairness involving multiple attributes and their proxies.

Intersectional fairness. Intersectional fairness has received increasing attention as researchers recognize that optimizing average fairness metrics can mask severe disparities for minority subgroups. Several works propose worst-group or max-gap objectives that explicitly optimize for the most disadvantaged group [9]. While these approaches improve fairness at the outcome level, they often do not directly address sensitive attribute information encoded in intermediate representations, which can still propagate bias to downstream tasks.

Bias in text embeddings and language models. Bias in pretrained language models and their embeddings has been extensively documented. Early studies demonstrated that word embeddings encode strong gender and racial stereotypes, even for words intended to be neutral [17, 6]. Subsequent work

proposed post-processing and fine-tuning methods to mitigate such biases, often relying on carefully constructed pairs of gendered words or projection-based techniques [17, 18]. While effective for textual data, these methods typically rely on domain-specific assumptions and do not readily generalize to other data modalities or to embedding-based decision making pipelines.

Limitations of prior work and our contribution. Most existing debiasing methods either rely on explicit sensitive attributes, adversarial objectives, or carefully constructed contrasting examples, or focus primarily on fairness at the output level. In contrast, our work addresses fairness in general embedding-based decision making models by proposing IterativeSifting, an iterative representation learning framework that progressively identifies and removes sensitive attribute information and proxy features. Unlike prior approaches, IterativeSifting naturally supports multiple sensitive attributes and intersectional groups, and does not require paired examples or modality-specific assumptions.

3 Gender Bias in Text Embeddings and a Baseline Mitigation Method

Pretrained language models are widely used in natural language processing due to their ability to map text into dense vector representations, or embeddings, that support a variety of downstream tasks [4, 5]. These embeddings are often reused as fixed representations rather than trained end-to-end, making their properties particularly important. However, because they are learned from large-scale text corpora, such representations may encode societal and historical biases present in the data [6, 17]. This section examines how gender bias manifests in text embeddings and motivates the need for methods to identify and mitigate its effects.

3.1 Gender Bias in Pretrained Language Models

Pretrained language models and their associated embeddings are widely used as foundational components in modern natural language processing systems. Word- and sentence-level embeddings produced by these models capture rich semantic and syntactic information and are commonly reused across a variety of downstream tasks, including sentiment analysis, text classification, information retrieval, and as inputs to generative models. Because these embeddings are often treated as generic representations of meaning, it is typically assumed that words or phrases that are semantically gender-neutral—such as "*manager*", "*nurse*", "*babysitter*", "*champion*", or "*engineer*"—should not encode strong gender-related information.

However, prior work has shown that pretrained language models can reflect historical and cultural biases present in their training corpora, and similar effects can be observed in their learned embedding spaces [6, 17]. In particular,

when gender-related features are extracted from embeddings, many words that are intended to be neutral exhibit a significant association with either male or female attributes. These biases are not inherent to the meanings of the words themselves, but rather reflect patterns of representation and usage in large-scale text data. As a result, gender bias can be implicitly encoded in the geometry of the embedding space, influencing similarity relationships and downstream model behavior.

Such biases may lead to unintended or harmful consequences when embeddings are used in practice. For example, biased representations can affect the outputs of generative models, reinforce stereotypes in language generation, or influence predictions in downstream decision making tasks such as sentiment analysis, recommendation, or content moderation [24, 25]. These concerns motivate a careful examination of how gender bias manifests in embedding-based representations and how it can be identified and mitigated.

To study this phenomenon empirically, I experimented with **sentence transformers** [26], a popular class of pretrained language models designed to produce high-quality word and sentence embeddings. My analysis reveals that many words and phrases that are expected to be gender-neutral nonetheless exhibit substantial gender bias in their embedding representations. In the following subsection, I introduce a method based on **sparse autoencoders (SAEs)** [3] to identify gender-related features in sentence transformer embeddings, quantify the associated bias, and explore baseline approaches for mitigating its effects.

3.2 Disentangling Gender Features and Mitigating Bias: A Baseline Method

In this subsection, we first describe how SAEs can be used to separate out the gender feature and to mitigate the gender bias. Then we will present some experimental results on sentence transformers.

3.2.1 Applying Sparse Autoencoders (SAEs) to Identify the Gender Feature and to Mitigate Bias

Sparse autoencoders (SAEs) are a class of neural networks designed to learn interpretable and structured representations of high-dimensional data. As illustrated in Figure 1, an SAE consists of an **encoder** that maps an input vector to a latent representation and a **decoder** that reconstructs the original input from this representation. Unlike standard autoencoders, SAEs impose a **sparsity constraint** on the latent layer, encouraging only a small subset of latent units to be active for any given input. This constraint promotes the learning of localized and semantically meaningful features, as each latent unit is encouraged to respond to a specific pattern in the data rather than distributing information across many dimensions.

Because of this sparsity property, SAEs have been used as a tool for **feature discovery and disentanglement** in learned representations. When applied to embeddings produced by a pretrained encoder, an SAE can be viewed as a

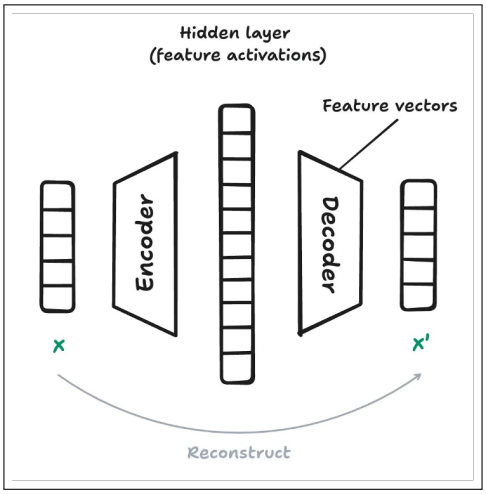


Figure 1: Illustrating the SAE model.

secondary model that re-expresses the original embedding space in terms of a set of sparse, potentially interpretable latent features. In this setting, individual latent units may correspond to distinct semantic or syntactic attributes encoded in the original embeddings.

In this work, SAEs are used to analyze embeddings generated by sentence transformer models. By training an SAE on these embeddings, we aim to identify latent units that capture gender-related information and to study how such information is distributed across the embedding space. This provides a mechanism for isolating gender features and serves as a foundation for measuring and mitigating gender bias in pretrained language model embeddings.

Disentangling the Gender Feature

To disentangle the gender feature using an SAE, we perform the following steps:

1. Construct a dataset of paired gendered word variants representing the same underlying role or concept, differing primarily in gender (e.g., (*actor*, *actress*)).

2. Feed each word in the pair to a target pretrained language model to obtain its embedding representation.

3. Feed the resulting word embeddings to a sparse autoencoder (SAE), one embedding at a time, and denote the latent representation at the output of the SAE encoder by $Z \in \mathbb{R}^d$.

4. For each dimension $j \in \{1, \ldots, d\}$, compute the average absolute difference between the corresponding latent values of the male and female word embeddings across all pairs.

5. Identify the dimension in Z with the largest average absolute difference and treat it as the gender-related feature.

Intuitively, this procedure identifies the latent dimension that most consistently distinguishes gendered word pairs while controlling for semantic content.

Removing Bias from a Given Set of Words

For a word that is intended to be gender-neutral, such as *engineer*, historical biases present in the training data of pretrained language models may cause its embedding to exhibit a non-neutral activation along the gender-related dimension identified above. In practice, we observe that this dimension often associates such words with a particular gender (e.g., a male association for the word *engineer*). We consider two approaches for mitigating this bias in a given word embedding:

- *Fine-tuning the embedding model.* If the pretrained language model allows fine-tuning, we define an auxiliary loss on the SAE encoder output that penalizes deviations of the gender-related latent dimension from a target neutral value. In this work, the neutral value is defined as the midpoint between the average activations of male- and female-associated word embeddings along this dimension.

- *Manipulating SAE activations.* Alternatively, we directly modify the activation value of the gender-related latent dimension at the SAE encoder output for the given word embedding, setting it to the neutral value, and then pass the modified latent representation through the SAE decoder to obtain a neutralized embedding.

3.2.2 Experimental Results for SAE Baseline

To disentangle the gender feature using an SAE, we first construct a dataset of paired gendered word variants representing the same underlying role or concept, differing primarily in gender. This dataset is derived from two publicly available resources: the *ecmonsen gendered words* dataset [19] and the Wiktionary English words by genders [20]. As the target pretrained language model, we use a widely adopted sentence-transformer encoder [21]. In particular, the results presented below are based on the *multi-qa-mpnet-base-cos-v1* model.

Figure 2 shows the activation values of the gender-related latent feature, corresponding to the 218th dimension of the SAE encoder output. We observe that the distribution of activations for *male* words (shown in cyan blue) occupies a higher value range than that of *female* words (shown in orange). The figure also includes activation values for several example words, indicated by vertical lines. Some of these words are inherently gendered, such as *grandpa* (male), *grandma* (female), and *actress* (female), and their activations align with this expectation. In contrast, words that are intended to be gender-neutral nonetheless exhibit biased activations along this dimension. For example, the embeddings of *manager* and *engineer* are biased toward the male side, while *babysitter* and *stay-at-home*

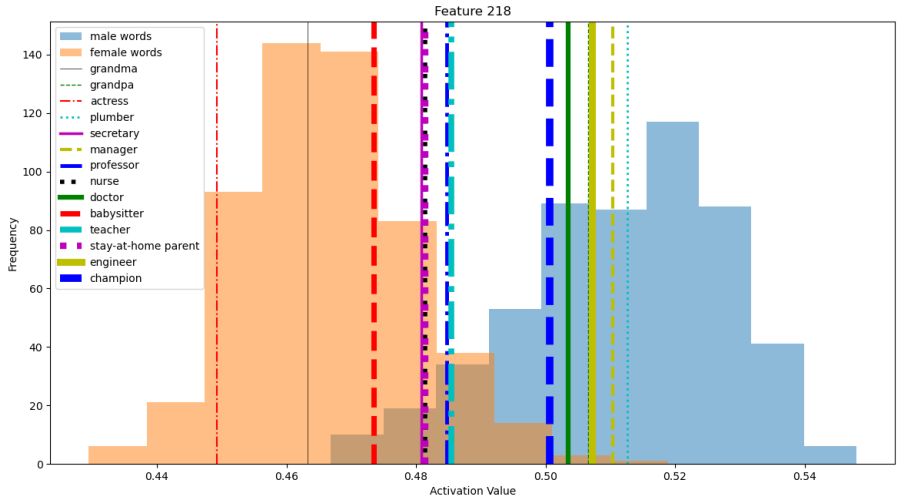


Figure 2: SAE shows the neuron activations of the gender feature.

parent are biased toward the female side, reflecting historical biases present in the training data of the pretrained language model.

We apply the fine-tuning approach described in Section 3.2.1 to mitigate the gender bias of a selected set of words. After fine-tuning, we obtain updated embeddings for these words and analyze them using the same SAE to examine the activation of the gender-related latent dimension. The results are shown in Figure 3. We observe that words whose gender bias was explicitly mitigated, such as *engineer*, *champion*, *stay-at-home parent*, and *nurse*, exhibit substantially neutralized activations along the gender dimension, while inherently gendered words such as *actress*, *grandpa*, and *grandma* remain largely unchanged. We emphasize that the SAE-based approach is primarily used as a baseline, serving as a point of comparison for the main method introduced in the following section.

4 Sensitive Attribute Information and Fairness in Embedding-Based Decision Models

The baseline SAE-based method presented in the previous section provides a useful starting point for identifying sensitive attribute information in embeddings. However, it relies on the availability of carefully constructed contrasting examples that represent the same underlying role or concept while differing primarily in a single sensitive attribute, as well as on the ability to isolate a

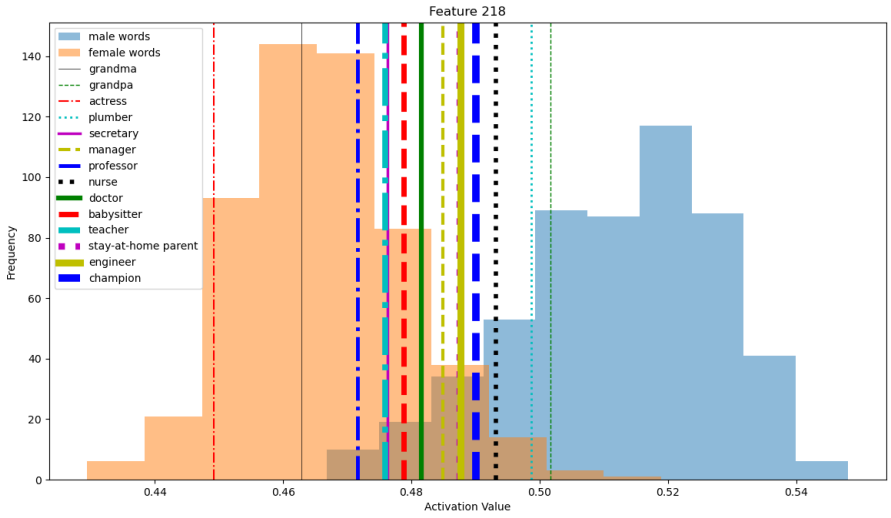


Figure 3: SAE shows the neuron activations of the gender feature after debiasing some words.

corresponding sensitive dimension in the SAE latent space. Although such assumptions may be reasonable in certain natural language settings, they are often violated in other data modalities, including tabular records or graph-structured data, where multiple sensitive attributes may be present and proxy variables can make sensitive information difficult to disentangle. In this section, we introduce our main method, termed *IterativeSifting*, which addresses fairness in general embedding-based decision making models and naturally extends to settings involving one or more sensitive attributes (e.g., gender, race, and age). Unlike the baseline approach, IterativeSifting does not require explicitly paired contrasting examples and operates directly on the learned embedding representations used by downstream decision models.

4.1 IterativeSifting: A Method for Fairness in Embedding-Based Decision Models

A typical machine learning pipeline for decision making consists of two main components. An encoder $\mathcal{F}$, such as a multilayer perceptron (MLP) [27] or a graph neural network (GNN) [28], maps the input data to a latent representation Z, also referred to as an embedding. The specific architecture of $\mathcal{F}$ depends on the data modality and task. A prediction head $\mathcal{H}$ then takes Z as input and produces the model output $\hat{Y}$.

The goal is to learn a representation that does not encode information re-

lated to one or more sensitive attributes (e.g., gender or race), including indirect proxy attributes that may serve as substitutes for sensitive information [29]. For example, geographic features such as zip code can act as proxies for race in certain decision making contexts [2]. At the same time, the learned representation should preserve task-relevant information necessary for accurate prediction of the target outcome, such as loan approval, job qualification, or income level. This tension between removing sensitive information and preserving predictive utility motivates the method introduced in this section. Since the learned representation (i.e., embedding) is designed to be free of sensitive attributes and their proxies, any downstream prediction based on this representation aims to be fair. This approach helps mitigate bias by ensuring the model cannot leverage spurious correlations.

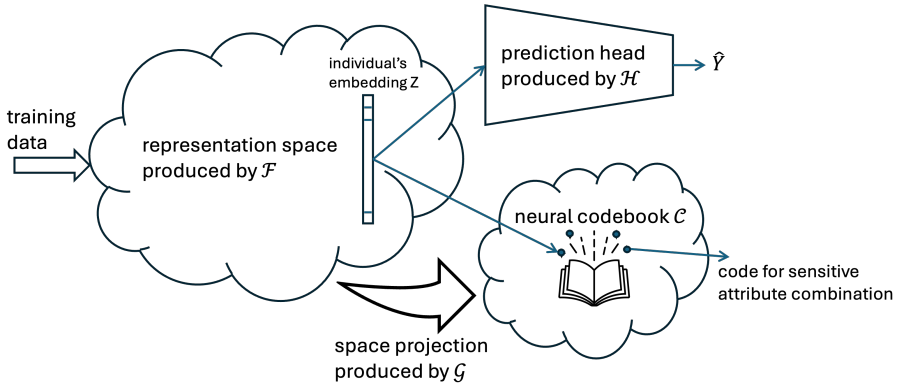


Figure 4: Illustrating the IterativeSifting method.

As illustrated in Figure 4, IterativeSifting consists of the following components:

- The training data is first mapped to a latent representation Z (i.e., embeddings) by an encoder $\mathcal{F}$, which may take the form of a multilayer perceptron (MLP), a convolutional neural network (CNN), or a graph neural network (GNN), depending on the data modality.
- A neural network, referred to as the *prediction head* $\mathcal{H}$, operates on the representation Z to produce the prediction $\hat{Y}$ for the target task.
- A *projection function* $\mathcal{G}$, implemented as a neural network, maps the embeddings Z into a projected embedding space that is used for fairness-aware processing.
- A *codebook* $\mathcal{C}$ is jointly learned in the projected embedding space, where each code $c \in \mathcal{C}$ corresponds to a specific joint value of one or more sensitive attributes (e.g., *White Male*). Each code c is itself represented as an embedding vector in the projected space.

These components interact through an iterative procedure that progressively reduces sensitive attribute information in the learned representations while preserving predictive performance. The goal of the *IterativeSifting* algorithm is to remove latent features and proxy information related to one or more sensitive attributes from the embeddings Z, using the projected embedding space and the codebook, while retaining task-relevant features necessary for accurate prediction by the head $\mathcal{H}$. The training procedure is summarized below.

1. **Joint initialization.** Jointly train the encoder $\mathcal{F}$, prediction head $\mathcal{H}$, projection function $\mathcal{G}$, and codebook $\mathcal{C}$ via back-propagation. The loss for $\mathcal{H}$ is task-dependent (e.g., binary cross-entropy for binary classification). The codebook $\mathcal{C}$ contains a learned embedding vector for each combination of sensitive attribute values. The loss for $\mathcal{G}$ encourages each sample's projected embedding to be close to the code corresponding to its sensitive attribute combination, while the loss for $\mathcal{C}$ enforces a margin that separates distinct codes in the projected space.

2. **Sifting step.** Freeze the parameters of $\mathcal{G}$ and the codebook $\mathcal{C}$, and update the encoder $\mathcal{F}$ and prediction head $\mathcal{H}$. In addition to the task loss for $\mathcal{H}$, an auxiliary loss is introduced that encourages the distances between a sample's projected embedding and all codes in $\mathcal{C}$ to be approximately equal. This step performs the *sifting* operation, reducing sensitive attribute–dependent information and proxy signals in the embeddings produced by $\mathcal{F}$.

3. **Refinement step.** Freeze the parameters of the encoder $\mathcal{F}$ (and hence the embeddings Z), and update $\mathcal{H}$, $\mathcal{G}$, and the codebook $\mathcal{C}$ using the same loss functions as in Step 1. Intuitively, this step refines the projection space and codebook to better capture any remaining sensitive attribute information present in Z.

4. **Iteration.** Repeat Steps 2 and 3 until a stopping condition is met. Specifically, the procedure terminates when the accuracy of predicting sensitive attribute combinations, computed by assigning each projected embedding to its nearest code in $\mathcal{C}$, fails to improve for a fixed number of consecutive iterations.

This alternating optimization procedure can be interpreted as a minimax-style process, in which the projection and codebook attempt to expose sensitive information while the encoder progressively removes it [11].

4.1.1 Intuition Behind IterativeSifting

The key intuition behind IterativeSifting is that sensitive attribute information is encoded in the representation space through directions that allow the projected embedding to be distinguishable with respect to sensitive attribute

values. The projection function $\mathcal{G}$ and codebook $\mathcal{C}$ explicitly expose these directions by learning to map embeddings to codes associated with sensitive attribute combinations.

During the sifting step, the encoder $\mathcal{F}$ is trained to produce embeddings whose projections are approximately equidistant from all codes in the codebook. This enforces invariance with respect to the sensitive attributes, as no single code is preferred over others in the projected space. Consequently, features in the representation that act as direct or indirect proxies for sensitive attributes are suppressed.

The refinement step then updates the projection function and codebook to capture any remaining sensitive attribute information present in the current embeddings. Alternating between these two steps creates an iterative process in which sensitive attribute information is progressively identified and removed from the representation. At the same time, the prediction head $\mathcal{H}$ is continuously trained on the task objective, ensuring that task-relevant information is preserved.

4.1.2 Loss Functions for IterativeSifting

We formalize the objective functions used in the IterativeSifting algorithm. Let X denote the input data, Y the target label, and $S \in \mathcal{S}$ the (possibly multi-dimensional) sensitive attribute vector. The encoder $\mathcal{F}$ maps inputs to embeddings $Z \in \mathbb{R}^d$, the prediction head $\mathcal{H}$ maps Z to predictions $\hat{Y}$, and the projection function $\mathcal{G}$ maps Z to a projected embedding $\tilde{Z} \in \mathbb{R}^k$. A codebook $\mathcal{C} = \{c_s \in \mathbb{R}^k : s \in \mathcal{S}\}$ contains one learnable code for each sensitive attribute combination.

Task loss. The task-dependent prediction loss is defined as

$$\mathcal{L}_{\text{task}}(\mathcal{F}, \mathcal{H}) = \mathbb{E}_{(x,y)}\left[\ell\big(\mathcal{H}(\mathcal{F}(x)), y\big)\right], \tag{1}$$

where $\ell(\cdot, \cdot)$ is an appropriate loss function (e.g., cross-entropy for classification).

Projection–code alignment loss. To encourage projected embeddings to align with their corresponding sensitive attribute codes, we define

$$\mathcal{L}_{\text{align}}(\mathcal{F}, \mathcal{G}, \mathcal{C}) = \mathbb{E}_{(x,s)}\left[\|\mathcal{G}(\mathcal{F}(x)) - c_s\|_2^2\right]. \tag{2}$$

Code separation (margin) loss. To ensure that codes corresponding to different sensitive attribute combinations remain distinguishable, we impose a margin constraint:

$$\mathcal{L}_{\text{margin}}(\mathcal{C}) = \sum_{\substack{s,s' \in \mathcal{S} \\ s \neq s'}} \max(0,\ m - \|c_s - c_{s'}\|_2), \tag{3}$$

where $m > 0$ is a fixed margin hyperparameter.

Algorithm 1 IterativeSifting

Input: Training data $\{(x_i, y_i, s_i)\}_{i=1}^n$, sensitive attribute set $\mathcal{S}$
Input: Encoder $\mathcal{F}$, prediction head $\mathcal{H}$, projection function $\mathcal{G}$, codebook $\mathcal{C}$
Input: Hyperparameters $\lambda_{\text{align}}, \lambda_{\text{margin}}, \lambda_{\text{sift}}$
Output: Fairness-aware encoder $\mathcal{F}$ and predictor $\mathcal{H}$

Initialization: Jointly train $\mathcal{F}$, $\mathcal{H}$, $\mathcal{G}$, and $\mathcal{C}$ by minimizing

$$\mathcal{L}_1 = \mathcal{L}_{\text{task}} + \lambda_{\text{align}}\mathcal{L}_{\text{align}} + \lambda_{\text{margin}}\mathcal{L}_{\text{margin}}.$$

repeat
  Sifting step: Freeze $\mathcal{G}$ and $\mathcal{C}$; update $\mathcal{F}$ and $\mathcal{H}$ by minimizing

$$\mathcal{L}_2 = \mathcal{L}_{\text{task}} + \lambda_{\text{sift}}\mathcal{L}_{\text{sift}}.$$

  Refinement step: Freeze $\mathcal{F}$; update $\mathcal{H}$, $\mathcal{G}$, and $\mathcal{C}$ by minimizing

$$\mathcal{L}_3 = \mathcal{L}_{\text{task}} + \lambda_{\text{align}}\mathcal{L}_{\text{align}} + \lambda_{\text{margin}}\mathcal{L}_{\text{margin}}.$$

until Sensitive attribute prediction accuracy does not improve
return $\mathcal{F}, \mathcal{H}$

Sifting loss. During the sifting step, the encoder is trained to remove sensitive attribute information by encouraging the projected embedding to be approximately equidistant from all codes. This is achieved via

$$\mathcal{L}_{\text{sift}}(\mathcal{F}, \mathcal{G}, \mathcal{C}) = \mathbb{E}_x \left[\frac{1}{|\mathcal{S}|} \sum_{s \in \mathcal{S}} \left(\|\mathcal{G}(\mathcal{F}(x)) - c_s\|_2 - \bar{d}(x) \right)^2 \right], \tag{4}$$

where

$$\bar{d}(x) = \frac{1}{|\mathcal{S}|} \sum_{s \in \mathcal{S}} \|\mathcal{G}(\mathcal{F}(x)) - c_s\|_2 \tag{5}$$

denotes the average distance between the projected embedding and all codes.

Overall objectives by training phase. The losses used in each phase of IterativeSifting are summarized as follows:

$$\mathcal{L}_1 = \mathcal{L}_{\text{task}} + \lambda_{\text{align}}\mathcal{L}_{\text{align}} + \lambda_{\text{margin}}\mathcal{L}_{\text{margin}}, \tag{6}$$
$$\mathcal{L}_2 = \mathcal{L}_{\text{task}} + \lambda_{\text{sift}}\mathcal{L}_{\text{sift}}, \tag{7}$$
$$\mathcal{L}_3 = \mathcal{L}_{\text{task}} + \lambda_{\text{align}}\mathcal{L}_{\text{align}} + \lambda_{\text{margin}}\mathcal{L}_{\text{margin}}, \tag{8}$$

corresponding respectively to the joint initialization, sifting, and refinement steps of the algorithm. IterativeSifting is summarized in Algorithm 1.

4.2 Experimental Results for IterativeSifting

4.2.1 Datasets and Metrics

We primarily evaluate IterativeSifting on the *Adult Census Income* dataset [22], a standard benchmark widely used in fairness research. We also conduct experiments on a larger and more recent dataset, *ACSIncome* [23], and observe similar trends; results on this dataset are presented near the end of this section. The Adult Census Income dataset is derived from the 1994 U.S. Census and contains individual-level attributes such as education level, occupation, age, marital status, and native country. The prediction task is to determine whether an individual's income exceeds \$50K per year. In our experiments, we consider *gender* and *race* as sensitive attributes, enabling the evaluation of intersectional fairness across multiple sensitive attributes. For all experiments, we implement the encoder $\mathcal{F}$, the prediction head $\mathcal{H}$, and the projection function $\mathcal{G}$ as multilayer perceptrons (MLPs).

We evaluate fairness at both the representation and decision levels. At the representation level, we quantify the dependence between the learned embedding Z and the sensitive attributes using **mutual information** [30]. Lower mutual information values indicate greater invariance of the representation to sensitive attributes. At the decision level, we measure fairness using equalized odds [7] and report the **maximum equalized odds difference** across the four intersectional groups defined by gender and race. Specifically, for each group, we compute the true positive rate (TPR) and false positive rate (FPR), and report the maximum absolute difference between any pair of groups across both rates. Formally, this is defined as

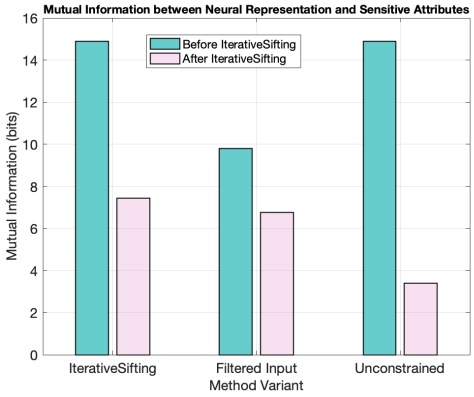
$$\max_{g,g'} \max(|\mathrm{TPR}_g - \mathrm{TPR}_{g'}|, |\mathrm{FPR}_g - \mathrm{FPR}_{g'}|).$$

This metric captures the worst-case disparity in model outcomes among intersectional subgroups. Model utility is measured using the prediction **accuracy** of the prediction head $\mathcal{H}$, reported both before and after applying the proposed debiasing algorithm to assess the impact on predictive performance. Together, these metrics allow us to evaluate how effectively IterativeSifting reduces sensitive attribute information in the representation while maintaining performance on the target task.

4.2.2 Evaluating IterativeSifting and Its Variants

We first evaluate IterativeSifting together with two variants to gain insight into the factors that contribute to its performance.

- *Filtered Input.* In this variant, we remove the explicit sensitive attributes *gender* and *race* from the input to the encoder $\mathcal{F}$. These attributes are still used to define sensitive attribute labels and to compute the fairness-related losses. This variant allows us to examine whether simply excluding sensitive attributes from the encoder input is sufficient to achieve fairness, or



(a) Mutual information comparison.

	IterativeSifting	Filtered Input	Unconstrained
Before	0.8587	0.8551	0.8587
After	0.8501	0.8527	0.2410

(b) Accuracy comparison.

Figure 5: Comparing variants of IterativeSifting.

whether proxy information for sensitive attributes remains in the learned representation.

- *Unconstrained.* In this variant, the prediction head $\mathcal{H}$ is trained only during the *Initialization* step of Algorithm 1. During the subsequent *Sifting* and *Refinement* steps, $\mathcal{H}$ is not updated and the task loss $\mathcal{L}_{\text{task}}$ is excluded from $\mathcal{L}_2$ and $\mathcal{L}_3$. This variant removes task-performance constraints during the debiasing process, allowing us to study the trade-off between fairness and predictive utility.

These variants serve as ablations that isolate the effects of input filtering and task constraints within IterativeSifting. We evaluate the mutual information between the encoder output embedding Z and the sensitive attributes, as well as the prediction accuracy of the prediction head, both before and after applying IterativeSifting, enabling direct comparison across the original method and its two variants. The results are shown in Figure 5.

The first two bars in Figure 5a show that the original IterativeSifting algorithm substantially reduces the mutual information between the encoder output Z and the sensitive attributes, decreasing it by approximately half from 14.9 to 7.46. This demonstrates the effectiveness of IterativeSifting in removing sensitive attribute information from the learned representation.

Interestingly, the second pair of bars shows that even after removing the explicit sensitive attributes *gender* and *race* from the encoder input, the mutual information between Z and the sensitive attributes S remains relatively high (nearly 10). This value is considerably larger than the mutual information obtained by applying IterativeSifting when sensitive attributes are included in the input, indicating that proxy attributes for S are present in the original dataset. After applying IterativeSifting to this filtered-input setting, the mutual information further decreases to a level below that of the original IterativeSifting

configuration. This result highlights the necessity of representation-level debiasing methods that go beyond simply removing sensitive attributes from the model input.

The last group of two bars in Figure 5a shows the result when IterativeSifting is applied without the constraint of training $\mathcal{H}$, the target prediction. We observe that the mutual information $I(Z;S)$ decreases further, dropping below 4. This demonstrates that aggressively removing sensitive attribute information from Z can conflict with retaining information useful for the target task.

Figure 5b compares the prediction accuracy across variants. Both the original IterativeSifting and the Filtered Input variant show only a slight accuracy decrease, indicating that IterativeSifting effectively preserves target prediction utility. In contrast, the unconstrained variant exhibits a significant accuracy drop, highlighting the inherent tradeoff between removing sensitive information and maintaining predictive performance.

Figure 6: Evaluation using the maximum equalized odds difference metric.

Next, we evaluate fairness using the *maximum equalized odds difference* metric, which measures disparities in true positive rates (TPRs) and false positive rates (FPRs) across the four intersectional sensitive attribute groups. The results are shown in Figure 6. We observe that disparities in TPRs are substantially larger than those in FPRs. In particular, before applying IterativeSifting, the *black female* group exhibits the lowest TPR, while the *white male* group has the highest TPR. After applying IterativeSifting, the *white female* group attains the highest TPR, and the overall gap among groups is reduced.

Importantly, the maximum equalized odds difference decreases by approximately half, from 0.222 to 0.115. This reduction indicates that IterativeSifting effectively mitigates worst-case disparities in model outcomes across intersectional groups. Combined with the reduction in mutual information, these results

suggest that removing sensitive attribute information from the representation translates into improved fairness at the decision level.

4.2.3 Comparisons with Baselines

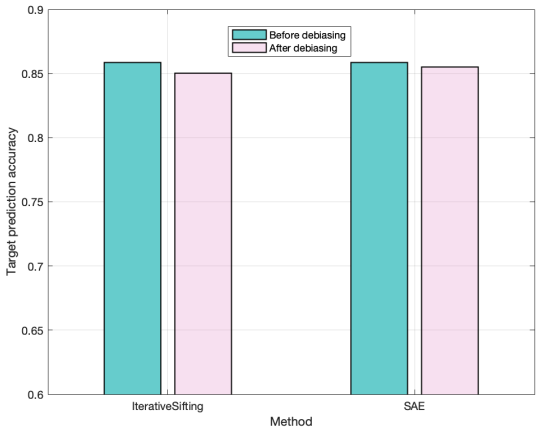
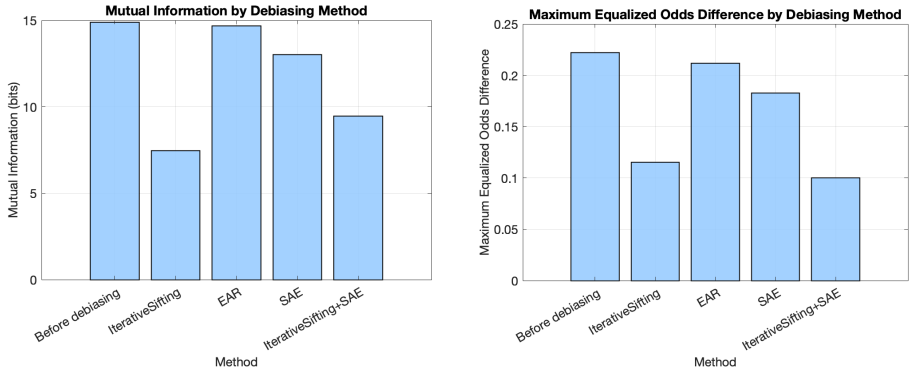


Figure 7: Comparing target prediction accuracy with the SAE baseline.

Now we compare IterativeSifting with baselines. A major baseline we compare against is the *sparse autoencoders* (SAEs) as presented in Section 3.2. We apply SAE to each of the two sensitive attributes and then neutralize the embeddings. First, we compare the accuracy of prediction head, as shown in Figure 7. In both cases, there is only a very slight decrease in prediction accuracy.

Next, we use the mutual information and the maximum equalized odds difference metrics. We also compare against another baseline called Editing with an Auxiliary Regression (EAR), which is essentially a gradient ascent–based approach in which a regression model predicts the sensitive attributes and gradient ascent is applied to erase sensitive features from the embeddings [31]. The results are shown in Figure 8. We can see that IterativeSifting achieves significantly better results than the two baselines. Even combining IterativeSifting with SAE yields only slightly better performance on the maximum equalized odds difference metric, while resulting in slightly worse performance on the mutual information metric.

The underlying reason is that, as discussed earlier, the SAE-based approach relies on the availability of carefully constructed contrasting examples that represent the same underlying concept while differing primarily in a single sensitive attribute. Although such pairs can be obtained for certain data types, such as natural language text, constructing them is challenging and often infeasible for other data modalities, including tabular data and graphs. As a result, the SAE approach does not perform as effectively in these settings.



(a) MI comparison with baselines.

(b) Fairness comparison with baselines.

Figure 8: Comparing IterativeSifting with baselines.

ACSIncome Dataset

We also experimented with a much larger dataset, the ACSIncome data [23]. We observe similar trends as those of the Adult Census Income dataset. Some of the results are shown in Table 1.

	Mutual information	**Max equalized odds diff.**
Without IterativeSifting	3.753	0.117
With IterativeSifting	2.317	0.062

Table 1: Mutual information and fairness comparisons using the ACSIncome dataset.

5 Conclusions and Future Work

In this work, we studied fairness in embedding-based machine learning models, with a particular focus on intersectional fairness in decision making tasks. We first examined gender bias in text embeddings produced by pretrained language models and introduced a baseline approach using sparse autoencoders to disentangle a gender-related feature and mitigate bias at the embedding level. While this approach provides useful insights into how sensitive attributes can be encoded in latent representations, it relies on carefully constructed contrasting examples and is therefore limited in its applicability to certain data modalities.

To address these limitations, we proposed *IterativeSifting*, a general and model-agnostic framework for reducing sensitive attribute information and proxy features in learned embeddings while preserving task-relevant information. IterativeSifting operates through an iterative training procedure that alternates between identifying sensitive attribute signals and removing them from the representation. Experimental results on standard fairness benchmarks demonstrate

that IterativeSifting substantially reduces mutual information between embeddings and sensitive attributes, improves intersectional fairness as measured by maximum equalized odds difference, and maintains competitive predictive accuracy.

There are several directions for future work. First, while we focused on tabular data and simple neural architectures in our experiments, IterativeSifting can be extended to more complex models and data modalities, such as graph neural networks and multimodal representations. Second, more sophisticated stopping criteria and convergence analyses could further improve the stability and efficiency of the iterative training process. Third, future work may explore integrating IterativeSifting with other fairness objectives or constraints, as well as evaluating its effectiveness under different definitions of fairness [32]. Finally, studying the behavior of IterativeSifting in real-world deployment settings, where sensitive attribute labels may be incomplete or noisy, remains an important direction for future research [10].

Beyond representation-level fairness, we also plan to study bias and fairness in LLM-based generative models. Since text generation is driven by latent representations, understanding how fairness and in particular intersectional fairness can be enforced during generation is an important problem.

References

[1] M. Kearns, S. Neel, A. Roth, and Z. S. Wu, *An empirical study of rich subgroup fairness for machine learning*, Proceedings of the Conference on Fairness, Accountability, and Transparency (FAT*), 2019.

[2] S. Barocas and A. D. Selbst, *Big data's disparate impact*, California Law Review, vol. 104, no. 3, pp. 671–732, 2016.

[3] M. Ranzato, Y. Boureau, and Y. LeCun, *Sparse feature learning for deep belief networks*, Advances in Neural Information Processing Systems (NeurIPS), 2007.

[4] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, *BERT: Pre-training of deep bidirectional transformers for language understanding*, Proceedings of NAACL-HLT, 2019.

[5] T. Brown et al., *Language models are few-shot learners*, Advances in Neural Information Processing Systems (NeurIPS), 2020.

[6] A. Caliskan, J. J. Bryson, and A. Narayanan, *Semantics derived automatically from language corpora contain human-like biases*, Science, vol. 356, no. 6334, pp. 183–186, 2017.

[7] Moritz Hardt, Eric Price, and Nati Srebro. Equality of Opportunity in Supervised Learning. *Advances in Neural Information Processing Systems (NeurIPS 2016)*. 2016.

[8] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. Fairness Through Awareness. *3rd Innovations in Theoretical Computer Science Conference (ITCS 2012)*. 2012.

[9] Michael Kearns, Seth Neel, Aaron Roth, and Zhiwei Steven Wu. Preventing Fairness Gerrymandering: Auditing and Learning for Subgroup Fairness. *35th International Conference on Machine Learning (ICML 2018)*. 2018.

[10] Joy Buolamwini and Timnit Gebru. Gender Shades: Intersectional Accuracy Disparities in Commercial Gender Classification. *Conference on Fairness, Accountability, and Transparency (FAT* 2018)*. 2018.

[11] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-Adversarial Training of Neural Networks. *Journal of Machine Learning Research (JMLR)*. 2016.

[12] Brian Hu Zhang, Blake Lemoine, and Margaret Mitchell. Mitigating Unwanted Biases with Adversarial Learning. *AAAI Conference on Artificial Intelligence (AAAI 2018)*. 2018.

[13] David Madras, Elliot Creager, Toniann Pitassi, and Richard Zemel. Learning Adversarially Fair and Transferable Representations. *35th International Conference on Machine Learning (ICML 2018)*. 2018.

[14] Zhewei Wei, Yao Ma, Jiliang Tang, and Suhang Wang. Adversarial Learning on Graphs: A Survey. *IEEE Transactions on Knowledge and Data Engineering (TKDE)*. 2021.

[15] Daniel Moyer, Shuyang Gao, Rob Brekelmans, Geert van den Broeck, and Stephan Mandt. Invariant Representations without Adversarial Training. *Advances in Neural Information Processing Systems (NeurIPS 2018)*. 2018.

[16] Alessandro Achille and Stefano Soatto. Information Dropout: Learning Optimal Representations through Noisy Computation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*. 2018.

[17] Tolga Bolukbasi, Kai-Wei Chang, James Zou, Venkatesh Saligrama, and Adam Kalai. Man is to Computer Programmer as Woman is to Homemaker? Debiasing Word Embeddings. *Advances in Neural Information Processing Systems (NeurIPS 2016)*. 2016.

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

[19] Ecmonsen gendered words. https://github.com/ecmonsen/gendered_words/.

[20] The English-language Wiktionary. https://en.wiktionary.org/wiki/Wiktionary:Main_Page.

[21] The sentence transformers. https://huggingface.co/sentence-transformers.

[22] The Adult Census Income dataset. https://archive.ics.uci.edu/dataset/2/adult.

[23] The ACSIncome dataset. https://fairlearn.org/main/user_guide/datasets/acs_income.html.

[24] E. Sheng, K.-W. Chang, P. Natarajan, and N. Peng, *The woman worked as a babysitter: On biases in language generation*, Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2019.

[25] S. L. Blodgett, S. Barocas, H. Daumé III, and H. Wallach, *Language (technology) is power: A critical survey of "bias" in NLP*, Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL), 2020.

[26] Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP 2019)*. 2019. URL: https://arxiv.org/abs/1908.10084.

[27] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. Deep Learning. *MIT Press*. 2016.

[28] Thomas N. Kipf and Max Welling. Semi-Supervised Classification with Graph Convolutional Networks. *5th International Conference on Learning Representations (ICLR 2017)*. 2017.

[29] Rich Zemel, Yu Wu, Kevin Swersky, Toniann Pitassi, and Cynthia Dwork. Learning Fair Representations. *Proceedings of the 30th International Conference on Machine Learning (ICML 2013)*. 2013.

[30] Mohamed Ishmael Belghazi, Aristide Baratin, Sai Rajeswar, Sherjil Ozair, Yoshua Bengio, Aaron Courville, and R. Devon Hjelm. Mutual Information Neural Estimation. *Proceedings of the 35th International Conference on Machine Learning (ICML 2018)*. 2018.

[31] Yanai Elazar and Yoav Goldberg. Adversarial Removal of Demographic Attributes from Text Data. *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP 2018)*. 2018.

[32] Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. A Survey on Bias and Fairness in Machine Learning. *ACM Computing Surveys*, 54(6). 2021.