Probing Classifiers are Unreliable for Concept Removal and Detection

Anonymous Author(s)
Affiliation
Address
email

Abstract

Neural network models trained on text data have been found to encode undesired linguistic or sensitive attributes in their representation. Removing such attributes is non-trivial because of a complex relationship between the attribute, text input, and the learnt representation. Recent work has proposed post-hoc and adversarial methods to remove such unwanted attributes from a model's representation. Through an extensive theoretical and empirical analysis, we show that these methods can be counter-productive: they are unable to remove the attributes entirely, and in the worst case may end up destroying all task-relevant features. The reason is the methods' reliance on a probing classifier as a proxy for the attribute. Even under the most favorable conditions when an attribute's features in representation space can alone provide 100% accuracy for learning the probing classifier, we prove that post-hoc or adversarial methods will fail to remove the attribute correctly. These theoretical implications are confirmed by empirical experiments on models trained on synthetic, Multi-NLI, and Twitter datasets. For sensitive applications of attribute removal such as fairness, we recommend caution against using these methods and propose a spuriousness metric to gauge the quality of the final classifier.

1 Introduction

Neural network classifiers built using text data have been shown to learn spuriously correlated features [14, 21] or embed sensitive attributes like gender or race [8, 7, 6] at their representation layer. Classifiers that use such sensitive or spurious concepts (henceforth *concepts*) raise concerns of model unfairness and out-of-distribution generalization [31, 3, 13]. Because the classifiers are based on hard-to-interpret deep neural networks, removing the influence of these concepts is non-trivial. Since many concepts cannot be modified at the input tokens level, removal methods that work at the representation layer have been proposed: post-hoc removal [7, 41, 12] on a pre-trained model (e.g., null space projection [25]), and adversarial removal [13, 40, 10] by jointly training the main task classifier with an (adversarial) classifier for the concept.

In this paper, we theoretically show that both classes of methods can be counter-productive in real-world settings where the main task label is often correlated with the concept. Examples include natural language inference (main task) where the presence of negative words (spurious feature) may be correlated with the "contradicts" label; or tweet sentiment classification (main task) where the author's gender (sensitive feature) may be correlated with the sentiment label. Our key result is based on the observation that both these methods internally use an auxiliary (or probing) classifier [1, 33] that aims to predict the spurious feature based on the representation learnt by the main classifier.

We show that the auxiliary classifier cannot be a reliable signal on whether the representation includes features that are causally derived from the concept. As previous work has argued [4, 35, 32, 3], if the representation features causally derived from the concept are not predictive enough, the probing classifier for the concept can be expected to rely on correlated features to obtain a higher accuracy.

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

However, we show a stronger result by considering a setting where there is no potential accuracy gain and the concept's features are easily learnable. Namely, even when the concept's causally-related features alone can provide 100% accuracy and are linearly separable from other features, the probing classifier may still learn non-zero weights for the main-task relevant features. Based on this result, under some simplifying assumptions, we prove that both post-hoc and adversarial training methods can fail to remove the undesired concept, remove useful task-relevant features in addition to the undesired concept, or do both As an extreme case, we show how post-hoc removal methods can lead to a random main-task classifier, removing all task-relevant information from the representation.

Empirical results on four datasets—natural language inference, sentiment analysis, tweet-mention detection and synthetic-text classification—confirm our claims. Across all datasets, as the correlation between the main task and the concept increases, we find that null space removal removes a higher amount of the main-task features, eventually leading to a random-guess classifier. In particular, for a pre-trained classifier that does not use the concept at all, the method modifies the representation to yield a lower main-task accuracy, irrespective of the correlation between the main task and the concept. For the adversarial removal method, we find a similar result that adversarial classifier does not remove all concept-related features. For most datasets, the concept-related features left within the adversarial classifier's representation is comparable to that for a standard main-task classifier. For practical usage of the removal methods, we propose a spuriousness metric that can be used to gauge the quality of the final classifier.

Our theoretical analysis complements empirical critique of the adversarial removal method [10]. It also extends the literature on probing classifiers and their unreliability [4]. Adding to known limitations of standard methods [16, 28] that evaluate accuracy of a probing classifier, our results show that recent causally-inspired methods like amnesic probing [11] are also flawed because they depend on access to a good quality classifier for the concept. Our contributions include:

- Theoretical analysis of null space and adversarial removal methods showing that they fail to remove an undesirable concept from a model's representation, even under favorable conditions.
- Empirical results on four datasets showing that the methods are unable to remove concept-related features fully and end up unnecessarily removing task-relevant features.
- A practical spuriousness score motivated by theory for evaluating the output of removal methods.

2 Concept removal: Background and problem statement

For a classification task, let $(\boldsymbol{x}^i, y^i_m)_{i=1}^n$ be set of examples in the dataset $\mathcal{D}_m$, where $\boldsymbol{x}^i \in \boldsymbol{X}$ is the input feature and $y^i_m \in Y_m$ the label. We call this as the *main task* and label y^i_m as the main task label. The main task classifier can be written as $c_m(h(\boldsymbol{x}))$ where $h : \boldsymbol{X} \to \boldsymbol{Z}$ is an encoder mapping the input $\boldsymbol{X}$ to the latent representation $\boldsymbol{Z}$ and $c_m : \boldsymbol{Z} \to Y_m$ is the classifier on top of the representation $\boldsymbol{Z}$. Additionally, we are given labels for a spurious or sensitive concept, $y^i_p \in Y_p$, and our goal is to ensure that the representation $h(\boldsymbol{x})$ learnt by the main classifier does not include features causally derived from the concept. Below we define what it means to be "causally derived": the representation should not change under an intervention on concept.

Definition 2.1. (**concept-causal feature**) A feature $Z_j \in \boldsymbol{Z}$ (jth dimension of $h(\boldsymbol{x})$) at the representation layer is said to be causally derived from a concept (concept-causal for short) if upon changing the value of the concept label, the corresponding change in the input's value $\boldsymbol{x}$ will lead to a change in the feature's (Z_j) value.

A concept removal algorithm is said to be successful if it produces a *clean* representation $h'(\boldsymbol{x})$ that has no concept-causal features to be used by the main-classifier and it does not corrupt or remove the main-task features. In practice, it is okay if the concept-causal features are not completed removed, but our key criterion is that the removal process should not remove the correlated main-task features. NEW (og88)

Existing concept removal methods: Adversarial training and null space projection. When the text input can be changed based on changing the value of concept label, methods like data augmentation [18] have been proposed for concept removal. However, for most sensitive or spurious properties, it is not possible to know the correct change to apply at the input level corresponding to a change in the property's value. Hence, the above approaches cannot be applied directly.

Instead, methods based on the representation layer have been proposed. To determine whether features in a representation are causally derived by the concept, these methods train an auxiliary, probing classifier $c_p : \boldsymbol{Z} \rightarrow Y_p$ where $y_p^i \in Y_p$ is the label of the concept we want to remove from the latent space $\boldsymbol{z}^i \in \boldsymbol{Z}$. The accuracy of the classifier denotes the predictive information about the concept embedded in the representation. This probing classifier is then used to remove the sensitive concept from the latent representation which will ensure that the main-task classifier cannot use them. Two kinds of feature removal methods have been proposed: 1) post-hoc methods such as null space removal ([25]), where removal is done after the main-task classifier is trained; 2) adversarial methods that jointly train the main task with the probing classifier as the adversary (e.g., [13, 40, 26, 27]).

Empirical results, however, have cast doubt on whether these methods (in particular, adversarial) can fully remove the sensitive concept from the model's representation [10]. In this paper, we complement such empirical work through a theoretical analysis of both adversarial and post-hoc removal methods. We also empirically investigate the null space method, and extend the analysis on adversarial methods by characterizing the effect of correlation between the main task and concept labels.

3 Attribute removal using probing classifier can be counter-productive

As mentioned above, both removal methods internally use a probing classifier as a proxy for the concept's features. In Section 3.1, we start off by showing that for any classification task be it probing or main-task classification, it is difficult to learn a *clean* classifier which doesn't use any spuriously correlated feature (Lemma 3.1 and Lemma C.5). Hence the key assumption driving the use of predictive classifiers within both removal methods is incorrect. Next in Section 3.2 and 3.3, we will show how these individual components' failure leads to overall failure of both removal methods. Finally, in Section 3.4, we propose a practical *spuriousness score* to assess the output classifier from the removal methods. Throughout this section, we assume that both the main task label y_m and probing task labels y_p is binary taking value from $\{-1, 1\}$ and there is a basic, fixed encoder h converting the text input to features in the representation space (e.g., it can be a pre-trained model like BERT).

3.1 Problem with learning a *clean* classifier: Main and Probing Classifier

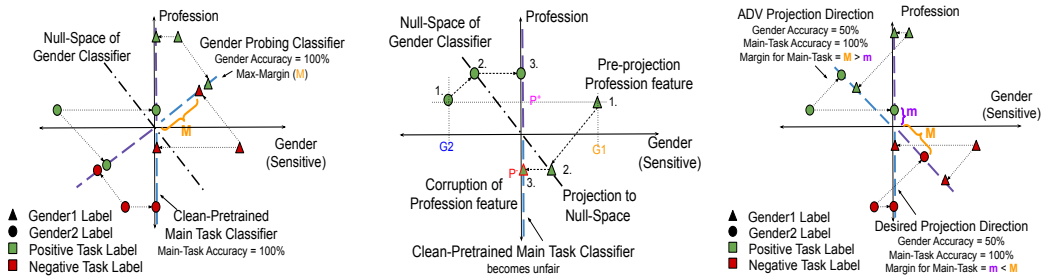
Given $\boldsymbol{z} = h(\boldsymbol{x})$ and the concept label y_p, the probing task is to learn a classifier $c_p(\boldsymbol{z})$ such that it only uses the concept-causal features and the accuracy for y_p is maximized. If there are features in z outside concept-causal that help improve the accuracy for the classifier, a classifier trained on standard losses such as cross-entropy or max-margin is expected to use those features too, as argued in the probing literature [16, 4]. In this section, however, we create a setting that is the most favorable for the probing classifier to use only concept-causal features—no accuracy gain on using features outside of concept-causal, disentangled representation so that no further representation learning is required, and class labels that are linearly separable using concept-causal features—yet we find that a trained classifier would use non-concept-causal features. Specifically, we make the following assumptions.

Assumption 3.1 (Disentangled Frozen Latent Representation)**.** The latent representation $\boldsymbol{z}$ is disentangled and is of form $[\boldsymbol{z}_m, \boldsymbol{z}_p]$, where $\boldsymbol{z}_p \in \mathbb{R}^{d_p}$ are the concept-causal feature and $\boldsymbol{z}_m \in \mathbb{R}^{d_m}$ are the features causally derived from the main task label along with rest of the features in latent representation. Here d_m and d_p are the dimensions of $\boldsymbol{z}_m$ and $\boldsymbol{z}_p$ respectively. Also the encoder $h(\boldsymbol{x})$ which maps the input $\boldsymbol{X}$ to latent representation $\boldsymbol{Z}$ is frozen or non-trainable.

Assumption 3.2 (concept-feature Linear Separability)**.** The concept-causal features of the latent representation ($\boldsymbol{z}_p$) for every point are linearly separable/fully predictive for the concept labels y_p, i.e $y_p^i \cdot (\hat{\boldsymbol{\epsilon}}_p \cdot \boldsymbol{z}_p^i + b_p) \geq 1$ for all datapoints $(\boldsymbol{x}^i, y_p^i)$ for some $\hat{\boldsymbol{\epsilon}}_p \in \mathbb{R}^{d_p}$. For the case of zero-centered latent space we have $b_p = 0$.

In addition, we assume that there is some correlation between the concept and main-task label: a function using z_m may also be able to classify correctly some non-empty subset of points.

Assumption 3.3 (Spurious Correlation)**.** For a subset of training points $\mathcal{S} \subset \mathcal{D}_p$ in probing dataset, z_m is linearly-separable with respect to concept label y_p i.e $y_p^i \cdot (\hat{\boldsymbol{\epsilon}}_m \cdot \boldsymbol{z}_m^i + b_m) \geq 1$ for some $\hat{\boldsymbol{\epsilon}}_m \in \mathbb{R}^{d_m}$ and $b_m \in \mathbb{R}$. For the case of zero-centered latent space we have $b_m = 0$.



(a) Clean Pretrained Classifier and Gender Classifier

(b) Main Classifier becomes unfair after null-space Projection

(c) Failure Mode of Adversarial Removal

Figure 1: **Failure Mode of Removal Methods**: For null-space removal see failure mode paragraph in Section 3.2 and for adversarial removal see failure mode paragraph in 3.3

Now we are ready to state the key lemma which will show that for only a *few special* points if non-concept-features $\boldsymbol{z}_m$ are linearly separable w.r.t. to concept label y_p, then the probing classifier $c_p(\boldsymbol{z})$ will use those features. Following [22], for ease of exposition we assume max-margin as training loss. Under some mild conditions on separable data, a classifier trained using logistic/exponential loss converges to max-margin classifier given infinite training time [34, 17].

Lemma 3.1. Let the latent representation be frozen and disentangled such that $\boldsymbol{z} = [\boldsymbol{z}_m, \boldsymbol{z}_p]$ (Assumption 3.1), where concept-features $\boldsymbol{z}_p$ be fully predictive (Assumption 3.2). Let $c_p^*(\boldsymbol{z}) = \boldsymbol{w}_p \cdot \boldsymbol{z}_p$ be the desired clean/purely-invariant linear classifier trained using max-margin objective (Appendix C.1) which only uses $\boldsymbol{z}_p$ for its prediction. Let $\boldsymbol{z}_m$ be the spurious feature s.t. for the margin points of $c_p^*(\boldsymbol{z})$, $\boldsymbol{z}_m$ be linearly-separable w.r.t. probing task label y_p (Assumption 3.3). Then, assuming the latent space is centered around $\mathbf{0}$, the concept-probing classifier trained using max-margin objective will be of form $c_p(\boldsymbol{z}) = \boldsymbol{w}_p \cdot \boldsymbol{z}_p + \boldsymbol{w}_m \cdot \boldsymbol{z}_m$ where $\boldsymbol{w}_m \neq \mathbf{0}$. FIX (PTq2)

Proof Sketch. We can perturb $c_p^*(\boldsymbol{z})$ to have $\boldsymbol{w}_m \neq \mathbf{0}$ and show that there always exist a perturbed classifier which have bigger margin than $c_p^*(\boldsymbol{z})$. For detailed proof see Appendix C.2. □

Our result shows that not just accuracy, even geometric skews in the dataset can yield an incorrect probing classifier. In Appendix C.3 we prove that the assumptions for Lemma 3.1 are both sufficient and necessary for a classifier to use non-concept-features $\boldsymbol{z}_m$ when $\boldsymbol{z}_p$ is 1-dimensional. Lemma 3.1 generalizes a result from [22] by using fewer assumptions (we do not restrict $\boldsymbol{z}_m$ to be binary, do not assume that $\boldsymbol{z}_m$ and $\boldsymbol{z}_p$ are conditionally independent given y, and do not assume monotonicity of classifier norm with dataset size). We present a similar result for main-task classifier (Lemma C.5) in Appendix C.4.

3.2 Failure mode of post-hoc removal methods: Null-space removal (INLP)

The null space method [25, 11], henceforth referred as *INLP*, removes a concept from latent space by projecting the latent space to a subspace that is not discriminative of that attribute. First, it estimates the subspace in the latent space discriminative of the concept we want to remove by training a probing classifier $c_p(\boldsymbol{z}^i) \rightarrow y_p^i$, where y_p^i is the concept label. Then the projection is done on to the null-space of this probing classifier which is expected to be non-discriminative of the concept. [25] used a linear probing classifier $(c_p(\boldsymbol{z}))$ to ensure that the any linear classifier cannot recover the removed concept from modified latent representation $\boldsymbol{z}'$ and hence the main task classifier $(c_m(\boldsymbol{z}'))$ also become invariant to removed attribute. Also, they recommend running this removal step for multiple iteration to ensure the unwanted concept is removed completely. Details of the method are in Appendix D.1.

Failure Mode: We use a 2-D latent feature representation to illustrate how null space removal can fail. One of the dimensions correspond to profession (main task) and the other gender. The input pre-trained classifier is clean, i.e., it uses only the profession feature for its prediction (Figure 1a). Next we apply the null-space removal to remove the *gender* concept. From Lemma 3.1, the obtained gender-probing classifier can also use *profession* feature, as shown by the projection direction—null-space of the gender classifier—being slanted (Figure 1a). As a result, Figure 1b shows that any two points (green points numbered 1) having the same *profession* feature but different gender feature, after projection to the null-space (slanted line, shown by number 2), will have their profession

feature reversed (shown by number 3). Even for a clean classifier, we see that null-space projection corrupts the task-specific features and thus the null-space projection step made a clean/fair classifier unclean/unfair. Note that the main-task classifier's accuracy will drop, affecting INLP based probing methods [11] that interpret a drop in main classifier's accuracy after projection as evidence that the sensitive concept was being used. Next we formalize these observations using $\boldsymbol{z}^{i(k)}$ to denote the representation z^i after k steps of INLP.

Theorem 3.2. (INLP Failure Mode). Let the latent representation $\boldsymbol{z}$ satisfies Assumption 3.1 and $\boldsymbol{c}_m(\boldsymbol{z})$ is the pre-trained main-task classifier and $\boldsymbol{c}_p(\boldsymbol{z})$ be the probing classifier used by INLP to remove the unwanted feature $\boldsymbol{z}_p$ from the latent representation. Let the assumptions of Lemma 3.1 be satisfied for the probing classifier $c_p(\boldsymbol{z})$ which is trained using max-margin objective. Then,

1. **Mixing:** After the first projection step of linear-INLP, the dimensions of $\boldsymbol{z}$ will get mixed such that $\boldsymbol{z}^{i(1)} = [g(\boldsymbol{z}_m^i, \boldsymbol{z}_p^i), f(\boldsymbol{z}_p^i, \boldsymbol{z}_m^i)] \neq [\boldsymbol{z}_m^i, \boldsymbol{z}_p^i]$. Also, this mixing is non-invertible with subsequent projection steps of INLP.

2. **Removal:** The L2-norm of the latent representation $\boldsymbol{z}$ decreases with every projection step as long as the probing classifier at a step does not lie completely in the null-space of previous probing classifiers. Thus, after sufficiently many steps, INLP can destroy all information in the representation, $\boldsymbol{z}^{i(\infty)} = [\mathbf{0}, \mathbf{0}]$.

Proof Sketch. From Lemma 3.1, probing classifier will have non-zero weight on the spurious feature $\boldsymbol{z}_m$. Hence, after projection operation they get mixed up and we show projection operator is non-invertible. Then we show projection operation is lossy i.e removes some norm of $\boldsymbol{z}$ when null-space of $c_p(\boldsymbol{z})$ is not orthogonal to $\boldsymbol{z}$. For detailed proof see Appendix D.2. □

3.3 Failure mode of adversarial removal methods

To remove the unwanted attribute $\boldsymbol{z}_p$ from the latent representation, the adversarial removal method jointly trains the main classifier $c_m : \boldsymbol{z} \rightarrow y_m$ and the probing classifier $c_p : \boldsymbol{z} \rightarrow y_p$ by specifying c_p's loss as an adversarial loss. For details refer to Appendix E.1.

Since the encoder $h : \boldsymbol{x} \rightarrow \boldsymbol{z}$ mapping the input to the latent representation $\boldsymbol{z}$ is frozen (Assumption 3.1), to allow training for the main-task classifier we introduce additional representation layers after it. For simplicity in the proof, we assume a linear transformation to the latent representation $h_2 : \boldsymbol{z} \rightarrow \boldsymbol{\zeta}$. After this, we have a linear main-task classifier $c_m : \boldsymbol{\zeta} \rightarrow y_m$, as before. The probing classifier $c_p : \boldsymbol{\zeta} \rightarrow y_p$ is trained adversarially to remove $\boldsymbol{z}_p$ from latent representation $\boldsymbol{\zeta}$. Thus, the goal of the adversarial method can be stated as removing the information of $\boldsymbol{z}_p$ from $\boldsymbol{\zeta}$. Let the main-task classifier satisfy the assumption of the generalized version of Lemma 3.1 (Appendix C.4). In addition we need an assumption on the hard-to-classify margin points which says that main-task labels and adversarial concept label are correlated on margin point of *clean* main-task classifier.

Assumption 3.4 (Label Correlation of Margin Point)**.** For the margin points of *clean* classifier for the main task, the adversarial-probing label y_p and the main task labels y_m are correlated, i.e w.l.o.g. $y_m^i = y_p^i$ for all margins points of clean main-task classifier.

Theorem 3.3. Let the latent representation $\boldsymbol{Z}$ satisfy Assumption 3.1, $h_2(\boldsymbol{z})$ be a linear transformation s.t. $h_2 : \boldsymbol{Z} \rightarrow \boldsymbol{\zeta}$, main classifier be $c_m(\boldsymbol{\zeta}) = \boldsymbol{w}_{c_m} \cdot \boldsymbol{\zeta}$ and the adversarial classifier be $c_p(\boldsymbol{\zeta}) = \boldsymbol{w}_{c_p} \cdot \boldsymbol{\zeta}$. Let all the assumptions of Lemma C.5 be satisfied for main-classifier $c_m(\cdot)$ when using $\boldsymbol{Z}$ directly as input and Assumption 3.2 be satisfied on $\boldsymbol{Z}$ w.r.t. adversarial-task. Let $h_2^*(\boldsymbol{z})$ be the desired encoder which is successful in removing $\boldsymbol{z}_p$ from $\boldsymbol{\zeta}$. Then there exists an undesired/incorrect encoder $h_2(\boldsymbol{z})$ such that $\boldsymbol{\zeta} = h_2(\boldsymbol{z}) = f(\boldsymbol{z}_m, \boldsymbol{z}_p)$ for some function "f" and the main classifier $c_m(h_2(\boldsymbol{z}))$ has bigger margin than $c_m(h_2^*(\boldsymbol{z}))$ and has,

1. $Accuracy(c_p(h_2(\boldsymbol{z})), y_p) = Accuracy(c_p(h_2^*(\boldsymbol{z})), y_p)$ when adversarial probing classifier $c_p(\cdot)$ is trained using any learning objective. Thus the undesired encoder $h_2(\boldsymbol{z})$ is indistinguishable from desired encoder $h_2^*(\boldsymbol{x})$ in terms of adversarial task prediction accuracy but better for main-prediction task in terms of max-margin objective.

2. $L(c_p(h_2(\boldsymbol{z})), y_p) > L(c_p(h_2^*(\boldsymbol{z})), y_p)$ under Assumption 3.4 where L is either max-margin or cross-entropy loss used for training $c_p(\cdot)$. Thus undesired encoder $h_2(\boldsymbol{z})$ is preferable over desired encoder $h_2^*(\boldsymbol{z})$ for both main and adversarial task objective.

Proof Sketch. First we show that w.l.o.g. $\boldsymbol{\zeta} \in \mathbb{R}$. Then the main classifier is same as $h_2(\zeta)$ and adversarial classifier is $\alpha \cdot h_2(\zeta)$ for some $\alpha \in \mathbb{R}$. Then we can show that given a clean h_2^* such that ζ dont contain sensitive $\boldsymbol{z}_p$, we can always perturb it to get $h_2 = \boldsymbol{w}_m \cdot \boldsymbol{z}_m + \boldsymbol{w}_p \cdot \boldsymbol{z}_p$ st. $\boldsymbol{w}_p \neq \mathbf{0}$. Then we show that this perturbed encoder has bigger margin for main classifier and equivalent accuracy or larger loss for adversarial classifier. For detailed proof see Appendix E.2. □

Failure Mode. We illustrate the result using the same 2D example. The goal is to remove the gender feature from the latent-representation shared by both gender and main-task (profession) classifier. Assume that the shared representation is a scalar value obtained by projecting the features in some direction. The adversarial goal is to find a projection direction such that gender classifier obtains a random guess accuracy 50%. As Figure 1c shows, there can be two such projection directions that yield 50% accuracy on gender. While both projection directions achieve same main-task accuracy, the slanted projection direction has bigger margin than the vertical direction and is thus preferred by main-task classifier trained using max-margin objective.

3.4 Extension to real-world data: A metric for quantifying degree Of spuriousness

Our theoretical analysis shows that probing-based removal methods fail to make the main task classifier invariant to unwanted concept. However, to verify whether the final classifier is using the concept or not, the theorem statements require knowledge of the concept's features $\boldsymbol{z}_p$. For practical usage, we propose a metric that quantifies the degree of failure or *spuriousness* for both the main and probing classifier. For simplicity, we define it assuming that both main task and concept are binary.

Let $\mathcal{D}_{m,p}$ be the dataset where for every input $\boldsymbol{x}^i$ we have both the main task label y_m and the concept label y_p. We define 2×2 groups, one for each combination of (y_m, y_p). Without loss of generality, assume that the main-task label $y_m = 1$ is spuriously correlated with concept label $y_p = 1$ and similarly $y_m = 0$ is correlated with $y_p = 0$. Thus, $(y_m = 1, y_p = 1)$ and $(y_m = 0, y_p = 0)$ are the majority group S_{maj} while groups $(y_m = 1, y_p = 0)$ and $(y_m = 0, y_p = 1)$ make up the minority group (S_{min}). We expect the main classifier to exploit this correlation and hence perform badly on S_{min} where the correlation breaks. Following [31], we posit that minority group accuracy i.e $Acc(S_{min})$ can be a good metric to evaluate the degree of *spuriousness*. We bound the metric by comparing it with the accuracy on S_{min} of a "clean" classifier that does not use the concept features.

Definition 3.1 (Spuriousness Score (ψ))**.** Given a dataset, $\mathcal{D}_{m,p} = S_{min} \cup S_{maj}$ containing binary task label and binary concept, let $Acc^f(S_{min})$ be the accuracy of the given main-task classifier (f) on the minority group S_{min} and $Acc^*(S_{min})$ be the accuracy on the minority group of a *clean* classifier that does not use the unwanted spurious concept. We define spuriousness score of "f" as:
$\psi(f) = \left|1 - \frac{Acc^f(S_{min})}{Acc^*(S_{min})}\right|$

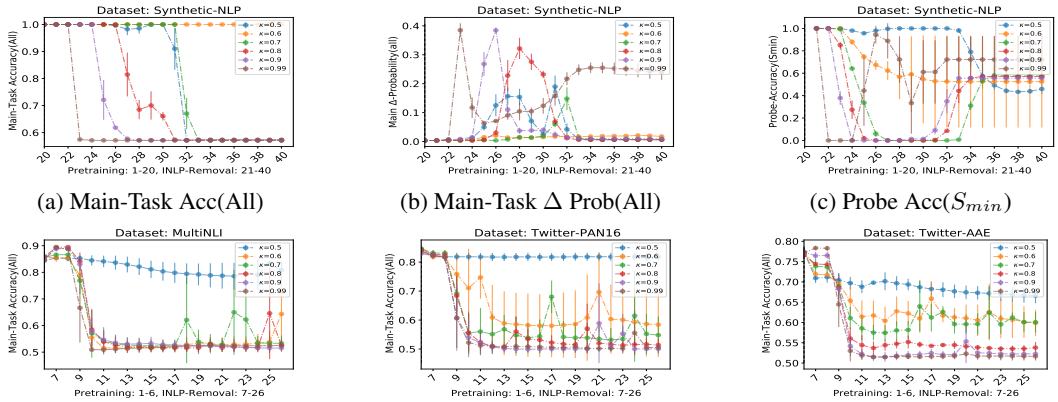
To estimate $Acc^*(S_{min})$, we subsample the dataset such that y_p takes a single value in the sample and train the main classifier on it, as done by [28]. Thus, probing label y_p no longer is correlated with the main-task label y_m. There can be other ways to estimate it, e.g., by reweighting the data or using the accuracy on S_{maj}. However, we found that the former had high variance and the latter requires an equal-noise assumption such that $Acc(S_{maj}) = Acc(S_{min})$ on a *clean* main-task classifier.

4 Experimental Results

Theorems 3.2 and 3.3 show the failure of concept removal methods under a simplified setup and max-margin loss. But current deep-learning models are not trained using max-margin objective and might not satisfy the required assumptions (Assumption 3.1,3.2,3.3,3.4). Thus, we now verify the failure modes on three real-world and one synthetic datasets, without making any restrictive assumptions. We use RoBERTa [20] as default encoder and fine-tune it over each real-world dataset. For Synthetic-Text dataset we use sum of pre-trained GloVe embeddings [23] of words in sentence as default encoder. For more details of experimental setup refer Appendix F.

4.1 Datasets: Main task and spurious/sensitive concept

Real-world data. We use three different real-world datasets: MultiNLI [38], Twitter-PAN16 [24] and Twitter-AAE [6]. In MultiNLI dataset, given two sentences—premise and hypothesis–the main task is to predict whether hypothesis *entails*, *contradicts* or is *neutral* with respect to premise. We simplify



(a) Main-Task Acc(All) (b) Main-Task Δ Prob(All) (c) Probe Acc(S_{min})

(d) Main-Task Acc(All) (e) Main-Task Acc(All) (f) Main-Task Acc(All)

Figure 2: **Null-Space Removal Failure**: For discussion see Section 4.2.

to a binary task of predicting whether a hypothesis *contradicts* the premise or not. Since negation words like *nobody,no,never* and *nothing* have been reported to be spuriously correlated with the *contradiction* label [15], We create a 'negation' concept denoting the presence of these words. The goal is to remove the negation property from the NLI model's representation space. In Twitter-PAN16 dataset, the main task is to detect whether a tweet mentions another user or not, as in [10]. The dataset contains *gender* label for each tweet, which we consider as the sensitive concept to be removed from the main model's representation. In Twitter-AAE dataset, again following [10], the main-task is to predict binary sentiment labels from a tweet's text. The tweets are associated with *race* of the author, the sensitive concept that should be removed from the main model's representation.

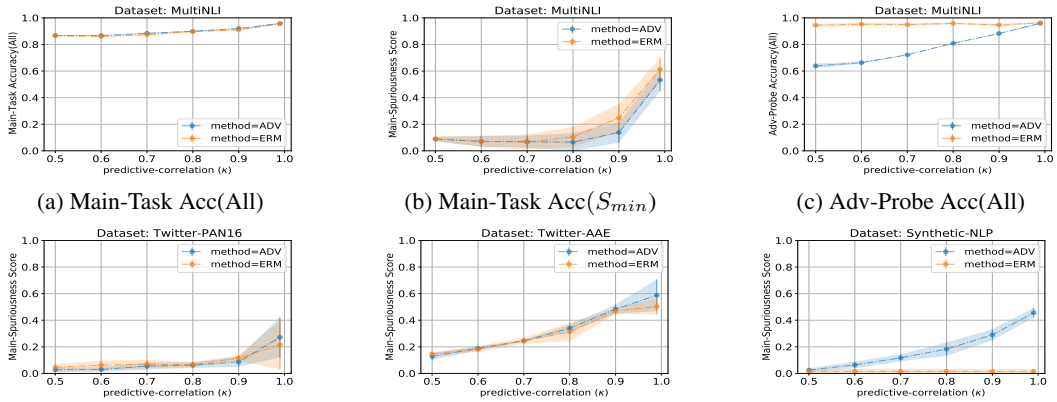
Synthetic-Text. To understand the reasons for failure, we introduce a Synthetic-Text dataset where it is possible to change the input text based on a change in concept (thus implementing Def. 2.1). Here we can directly evaluate whether the concept is being used by the main-task classifier by intervening on the concept (adding or removing) and observing the change in prediction scores. The main-task is to predict whether a sentence contains a numbered word (e.g., *one, fifteen, etc.*). We introduce spurious concept (length) by increasing the length of sentences that contain *numbered* words.

Predictive correlation. To assess robustness of removal methods, we create multiple datasets with different *predictive* correlation between two labels y_m and y_p. The predictive correlation (κ) measures how informative one variable is for predicting the other, $\kappa = Pr(y_m \cdot y_p > 0) = \frac{\sum_{i=1}^{N} \mathbf{1}[y_m \cdot y_p > 0]}{N}$, where N is the size of dataset and $\mathbf{1}[\cdot]$ is the indicator function which is 1 if the argument is true otherwise 0. Predictive-correlation lies in $\kappa \in [0.5, 1]$ where $\kappa = 0.5$ means no-correlation and $\kappa = 1$ means the attributes are completely correlated. For complete description of all datasets and how we vary the predictive-correlation, refer to Appendix F; and for additional results refer Appendix G.

4.2 Results: Null Space Removal

In general, for any model given as input to INLP, it may be difficult to verify whether the removal method removed the correct features. Hence, we construct a benchmark where the input classifier is *clean*, i.e., it does not use the concept at all. We do so by training on a subset of data with one particular value of spurious concept label, as in [28]. Since the input classifier does not have concept-causal features in it's representation, we should expect that INLP should not have any effect on the main task classifier. NEW (PTq2)

Eventually all task-relevant features are destroyed. We start with the Synthetic-Text dataset by training a clean classifier on the main-task and input it to INLP for removing the sensitive concept. To keep the conditions favorable for INLP, both the main task and sensitive-concept probing task can achieve 100% accuracy using their causally derived features respectively. In Figure 2a, iteration 21-40 shows individual steps of null-space removal. Since, the given pre-trained classifier was *clean* i.e not using the sensitive features, null-space removal shouldn't have any effect on it. Colored lines show different datasets used by INLP with different level of predictive correlation κ. We observe that for all value of κ, the main-task classifier's accuracy eventually goes to 50% random guess accuracy



(a) Main-Task Acc(All) (b) Main-Task Acc(S_{min}) (c) Adv-Probe Acc(All)

(d) Main-Task Acc(S_{min}) (e) Main-Task Acc(S_{min}) (f) Main-Task Acc(S_{min})

Figure 3: **Adversarial Removal Failure**: For discussion see Section 4.3.

implying that the main-task related attribute has been removed by the removal process, as predicted by Theorem 3.2. Higher the value of correlation κ, faster the removal of main-task attribute happens. We obtain a similar pattern over the real-world datasets, MultiNLI, Twitter-PAN16 and Twitter-AAE. Figure 2d,2e and 2f show a decrease in the main-task accuracy to random-guess even when the input classifier for each dataset is ensured to be *clean*, thus demonstrating the failure of INLP method.

Early stopping increases the reliance on spurious features. To avoid the full collapse, a stopping criteria of INLP is to stop when the main-task classifier's performance drops [25]. In Figure 2b we measure the sensitivity of Synthetic-Text main-task classifier w.r.t. to sensitive concept by changing the feature in input sentences corresponding to sensitive concept and measuring the change in main-task classifier prediction probability (ΔProb). At lower iterations of INLP, the change in main-task output due to change in sensitive concept's value, ΔProb, is higher than that of the input classifier. For example, for $\kappa = 0.8$, the main-task classifier's performance drops for the first time at iteration 27, but it has a high $\Delta\text{Prob} \approx 25\%$ as shown in Figure 2b. Hence it is possible that stopping prematurely will lead us to a classifier which is more unfair or reliant to sensitive concept than it was before, consistent with the first statement in Theorem 3.2 stating that INLP will lead to mixing of features in latent space.

Failure of causally-inspired probing like Amnesic probing. Amnesic Probing [11] declares that a sensitive concept is being used by the model if, after removal of the concept from from the latent representation using INLP, there is a drop in the main-task performance. But Figure 2a, 2d, 2e and 2f show that even when the input classifier for their corresponding main-task is clean, i.e., does not use the sensitive concept, INLP leads to drop in performance of the main-classifier. Hence, removal-based methods like Amnesic probing will falsely claim that the sensitive concept is being used.

4.3 Results: Adversarial Removal Method

We demonstrate the failure of adversarial-removal method (AR) in removing the spurious/unwanted concept from the main classifier. We train a separate main-task classifier without any adversarial objective with standard cross-entropy loss objective (referred as ERM henceforth) . Then we compare the standard ERM training of the main classifier with AR method over the same number of epochs (20). We follow the training procedure of [10], and conduct a hyper-parameter sweep on the adversarial training strength to select the value which is most effective in removing the unwanted property. For more details, refer to Appendix F. NEW (PTq2)

Cannot remove the spurious concept fully. For MultiNLI, Figure 3b shows the spuriousness score (Definition 3.1) of ERM and AR classifiers as we vary the predictive correlation (κ) between the main-task label and sensitive concept label in the training dataset. While the spuriousness score for classifier trained using AR (blue curve) is lower than that of ERM for all values of κ, it is substantially away from zero. Thus, the AR method fails to completely remove the sensitive/spurious concept completely from the latent representation. Figure 3c shows the possible reason using the sensitive concept probing classifier accuracy for ERM and AR. The probe accuracy after adversarial training

doesn't decrease to 50% but stops at accuracy proportional to the predictive correlation κ. This is expected since even if the AR would have been successful in removing the sensitive feature, the main-task feature would still be predictive of sensitive concept label by κ due to the spurious correlation between them. However, the converse is not true: an accuracy of κ does not imply that sensitive concept is fully removed. The results substantiate the first statement of Theorem 3.3: given two representations where one (*desired*) does not have concept features while the other (*undesired*) contains the sensitive concept features, the undesired one may be better for the main task loss even as both may have the same probing accuracy. Figure 3d, 3e and 3f show the spuriousness score of AR in comparison to classifier trained with ERM on Twitter-PAN16, Twitter-AAE and Synthetic-Text datasets respectively. The failure of AR is worse here: there is no significant reduction in spuriousness score for AR in comparison to ERM. In case of Synthetic-Text dataset, ERM has zero spuriousness score but AR have non-zero score. We expand more on this observation in Appendix G.3.

Comparison with previous work. Our experimental results extend the observations from previous work on adversarial removal failure [10]. If post-removal the latent representation used by the main-task classifier is still predictive of the removed concept, [10] claimed it as a failure of the removal method. However, this claim may not be correct since a feature could be present in the latent space and yet not used by model [28].Therefore, we propose the spuriousness score for the main classifier as a metric to verify if a feature is being used. We also confirm the observation from [10] that adversarial removal can fail even if there is no correlation between concept and task label ($\kappa = 0.5$), and extend their work by showing failure modes on multiple values of κ. For more details on verifying this observation with other metrics and other ablations, see Appendix G.3.

Ablations. In Appendix, we report results on using BERT instead of RoBERTa as the input encoder (Appendix G.2, G.3), the effect of using different modeling choices like loss-function, regularization, e.t.c. (G.4), and the behavior of probing classifiers when concept is not there in latent space (G.1).

5 Related work

Concept removal methods. When the removal of a concept can be simulated in input space (e.g., in tabular data or simpler concepts), approaches to remove a concept include data augmentation [18], gradient regularization [30]. However, when change in a concept cannot be propagated via change in input tokens, it becomes a non-trivial problem. Apart from the null space and adversarial removal, past work has used explanations for the classifier's prediction [19]. Methods like [26, 27] have tried to restrict the adversarial function to be projection operation like [25] and derive close form solution for concept removal for various objective. But all of them could be clubbed under same umbrella of adversarial removal. Given the difficulty of building a good estimator for the features causally derived from the concept, our work highlights the failure modes of popular removal methods.

Limitations of using a probing classifier for model interpretability. We also contribute to the growing literature on the limitations of a probing classifier's accuracy in capturing whether the main classifier is using the concept [4], It is known that probing classifiers capture not just the property, but any other features that may be correlated with it [31, 3, 18, 36]. As a result, many improvements have been proposed to better estimate whether a property is being used, including the use of control labels or datasets [16, 28]. Parallely, new causality-inspired probing methods [11] have been proposed that compare the main task accuracy on a representation without the property, constructed using the null space removal method. The hope is that such improvements can make probing more robust. Our results question this direction. To demonstrate the unreliability of a probing classifier, we construct a setup that is most favorable for learning only the property's features and still find that learned probing classifier will include a non-zero weight for the other features too. As we showed in Section 3.2 and 4.2, causally-inspired probing methods [11] have the same limitation because the removal method depends on the output of an auxiliary classifier that is equivalent to a probing classifier.

6 Conclusion and Discussion

Using both theoretical analysis and empirical results over real-world and synthetic datasets, we presented a critique of recent work that claims to remove spurious or sensitive concepts from a text classifier. Given the practical relevance of the removal problem, we also provided an spuriousness metric to evaluate any classifier on its use of unwanted concepts.

References

[1] Yossi Adi, Einat Kermany, Yonatan Belinkov, Ofer Lavi, and Yoav Goldberg. Fine-grained analysis of sentence embeddings using auxiliary prediction tasks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017.

[2] Abien Fred Agarap. Deep learning using rectified linear units (relu). *arXiv preprint arXiv:1803.08375*, 2018.

[3] Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization, 2020.

[4] Yonatan Belinkov. Probing Classifiers: Promises, Shortcomings, and Advances. *Computational Linguistics*, 48(1):207–219, 04 2022.

[5] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag, Berlin, Heidelberg, 2006.

[6] Su Lin Blodgett, Lisa Green, and Brendan O'Connor. Demographic Dialectal Variation in Social Media: A Case Study of African-American English. In *Proceedings of EMNLP*, 2016.

[7] 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. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, page 4356–4364, Red Hook, NY, USA, 2016. Curran Associates Inc.

[8] Alexis Conneau, German Kruszewski, Guillaume Lample, Loïc Barrault, and Marco Baroni. What you can cram into a single $&!#* vector: Probing sentence embeddings for linguistic properties. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2126–2136, Melbourne, Australia, July 2018. Association for Computational Linguistics.

[9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[10] Yanai Elazar and Yoav Goldberg. Adversarial removal of demographic attributes from text data. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 11–21, Brussels, Belgium, October-November 2018. Association for Computational Linguistics.

[11] Yanai Elazar, Shauli Ravfogel, Alon Jacovi, and Yoav Goldberg. Amnesic probing: Behavioral explanation with amnesic counterfactuals. *Trans. Assoc. Comput. Linguistics*, 9:160–175, 2021.

[12] Kawin Ethayarajh, David Duvenaud, and Graeme Hirst. Understanding undesirable word embedding associations. In Anna Korhonen, David R. Traum, and Lluís Màrquez, editors, *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 1696–1705. Association for Computational Linguistics, 2019.

[13] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*, ICML'15, page 1180–1189. JMLR.org, 2015.

[14] Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel Bowman, and Noah A. Smith. Annotation artifacts in natural language inference data. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 107–112, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.

[15] Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel Bowman, and Noah A. Smith. Annotation artifacts in natural language inference data. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 107–112, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.

[16] John Hewitt and Percy Liang. Designing and interpreting probes with control tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2733–2743, Hong Kong, China, November 2019. Association for Computational Linguistics.

[17] Ziwei Ji and Matus Telgarsky. Risk and parameter convergence of logistic regression, 2018.

[18] Divyansh Kaushik, Amrith Setlur, Eduard Hovy, and Zachary C Lipton. Explaining the efficacy of counterfactually augmented data. *International Conference on Learning Representations (ICLR)*, 2021.

[19] Sawan Kumar and Partha Talukdar. NILE : Natural language inference with faithful natural language explanations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8730–8742, Online, July 2020. Association for Computational Linguistics.

[20] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692, 2019.

[21] Tom McCoy, Ellie Pavlick, and Tal Linzen. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3428–3448, Florence, Italy, July 2019. Association for Computational Linguistics.

[22] Vaishnavh Nagarajan, Anders Johan Andreassen, and Behnam Neyshabur. Understanding the failure modes of out-of-distribution generalization. *ArXiv*, abs/2010.15775, 2021.

[23] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, 2014.

[24] Francisco Rangel, Paolo Rosso, Ben Verhoeven, Walter Daelemans, Martin Potthast, and Benno Stein. Pan16 author profiling, September 2016.

[25] Shauli Ravfogel, Yanai Elazar, Hila Gonen, Michael Twiton, and Yoav Goldberg. Null it out: Guarding protected attributes by iterative nullspace projection. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7237–7256, Online, July 2020. Association for Computational Linguistics.

[26] Shauli Ravfogel, Michael Twiton, Yoav Goldberg, and Ryan Cotterell. Linear adversarial concept erasure. *CoRR*, abs/2201.12091, 2022.

[27] Shauli Ravfogel, Francisco Vargas, Yoav Goldberg, and Ryan Cotterell. Adversarial concept erasure in kernel space. *CoRR*, abs/2201.12191, 2022.

[28] Abhilasha Ravichander, Yonatan Belinkov, and Eduard Hovy. Probing the probing paradigm: Does probing accuracy entail task relevance? In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 3363–3377, Online, April 2021. Association for Computational Linguistics.

[29] Radim Rehurek and Petr Sojka. Gensim–python framework for vector space modelling. *NLP Centre, Faculty of Informatics, Masaryk University, Brno, Czech Republic*, 3(2), 2011.

[30] Andrew Slavin Ross, Michael C. Hughes, and Finale Doshi-Velez. Right for the right reasons: Training differentiable models by constraining their explanations. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*, pages 2662–2670, 2017.

[31] Shiori Sagawa, Pang Wei Koh, Tatsunori B Hashimoto, and Percy Liang. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. In *International Conference on Learning Representations (ICLR)*, 2020.

[32] Shiori Sagawa, Aditi Raghunathan, Pang Wei Koh, and Percy Liang. An investigation of why overparameterization exacerbates spurious correlations. In *Proceedings of the 37th International Conference on Machine Learning*, ICML'20. JMLR.org, 2020.

[33] Xing Shi, Inkit Padhi, and Kevin Knight. Does string-based neural MT learn source syntax? In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1526–1534, Austin, Texas, November 2016. Association for Computational Linguistics.

[34] Daniel Soudry, Elad Hoffer, Mor Shpigel Nacson, Suriya Gunasekar, and Nathan Srebro. The implicit bias of gradient descent on separable data. *J. Mach. Learn. Res.*, 19(1):2822–2878, jan 2018.

[35] Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry. Robustness may be at odds with accuracy. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019.

[36] Victor Veitch, Alexander D'Amour, Steve Yadlowsky, and Jacob Eisenstein. Counterfactual invariance to spurious correlations in text classification. In Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan, editors, *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 16196–16208, 2021.

[37] Eric W Weisstein. "symmetric matrix." from mathworld–a wolfram web resource, https://mathworld.wolfram.com/symmetricmatrix.html.

[38] Adina Williams, Nikita Nangia, and Samuel Bowman. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122. Association for Computational Linguistics, 2018.

[39] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. Huggingface's transformers: State-of-the-art natural language processing, 2019.

[40] Qizhe Xie, Zihang Dai, Yulun Du, Eduard Hovy, and Graham Neubig. Controllable invariance through adversarial feature learning. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 585–596, Red Hook, NY, USA, 2017. Curran Associates Inc.

[41] Ke Xu, Tongyi Cao, Swair Shah, Crystal Maung, and Haim Schweitzer. Cleaning the null space: A privacy mechanism for predictors. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, AAAI'17, page 2789–2795. AAAI Press, 2017.

[A] Moritz Hardt, Eric Price, Eric Price, Nati Srebro; Equality of Opportunity in Supervised Learning; Advances in Neural Information Processing System, 2016

NEW (WqdQ, og88, PP9U, PTq2)

[B] Mehrabi, Ninareh and Morstatter, Fred and Saxena, Nripsuta and Lerman, Kristina and Galstyan, Aram; A Survey of Bias and Fairness in Machine Learning; Vol 54, No. 6, ACM Comp. Surv. Journal, 2021

[C] Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin and Sameer Singh; Beyond Accuracy: Behavioral Testing of NLP Models with CheckList; Association for Computational Linguistics, 2020

[D] Ananth Balashankar, Xuezhi Wang, Ben Packer, Nithum Thain, Ed Chi, Alex Beutel; Can We Improve Model Robustness through Secondary Attribute Counterfactuals? Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes] See Section 3 and 4
   (b) Did you describe the limitations of your work? [Yes] See Appendix B
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] See Appendix B
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes] Our work show failure modes of existing method.
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes] See Section 3
   (b) Did you include complete proofs of all theoretical results? [Yes] See Appendix C,D and E.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] See Appendix F.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Appendix F.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] See Section 4 and Appendix F.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See Appendix F.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] See Section 4 and Appendix F.
   (b) Did you mention the license of the assets? [Yes] See Appendix F.
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes] See Appendix F.
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [Yes] See Appendix F.
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [Yes] See Appendix F.
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]