Learning from Others: Similarity-based Regularization for Mitigating Dataset Bias

Reda Igbaria **Yonatan Belinkov**
Technion - Israel Institute of Technology
redaigbaria@technion.ac.il belinkov@technion.ac.il

Abstract

Common methods for mitigating spurious correlations in natural language understanding (NLU) usually operate in the output space, encouraging a main model to behave differently from a bias model by down-weighing examples where the bias model is confident. While improving out-of-distribution (OOD) performance, it was recently observed that the internal representations of the presumably debiased models are actually more, rather than less biased. We propose SimReg, a new method for debiasing internal model components via similarity-based regularization, in representation space: We encourage the model to learn representations that are either similar to an unbiased model or different from a biased model. We experiment with three NLU tasks and different kinds of biases. We find that SimReg improves OOD performance, with little in-distribution degradation. Moreover, the representations learned by SimReg are less biased than in other methods.[1]

1 Introduction

Recent studies (McCoy et al., 2019; Geirhos et al., 2020, *inter alia*) show that in many cases neural models tend to exploit spurious correlations (a.k.a dataset biases, artifacts[2]) in datasets and learn shortcut solutions rather than the intended function. For example, in MNLI—a popular Natural Language Understanding dataset—there is a high correlation between negation words such as "not, don't" and the *contradiction* label (Gururangan et al., 2018). Thus models trained on MNLI confidently predict contradiction whenever there is a negation word in the input without considering the whole meaning of the sentence. As a result of relying on such shortcuts, models fail to generalize and perform poorly when tested on out-of-distribution data (OOD) in which such associative patterns are not present (McCoy et al., 2019); these models are commonly known as 'biased' models. Moreover, this behavior limits their practical applicability in cases where the real-world data distribution differs from the training distribution.

Recent efforts to mitigate learning spurious correlations (a.k.a debiasing methods) perform the debiasing extrinsically, i.e., operating on the output space of the model and dictating how its output should look like. Typically, by downweigh the importance of training samples that contain such correlations, effectively performing data reweighting (Schuster et al., 2019; Utama et al., 2020a; Sanh et al., 2021; Cadene et al., 2019). One might expect that such an *extrinsic* debiasing would lead to "suppressing the model from capturing non-robust features" (Du et al., 2023). However, Mendelson and Belinkov (2021) showed a counter-intuitive trend: a higher accuracy of such models on OOD challenge sets is correlated with a higher representation bias,[3] i.e.,the more extrinsically de-biased a model is, the stronger its intrinsic bias. Such superficial debiasing is problematic as the bias may reappear when the model is used in another setting (Orgad et al., 2022), such as fine-tuned on more data or transferred to other similar tasks.

Inspired by this finding, we investigate whether debiasing the model intrinsically (i.e., operating in the representation space) leads to better models both extrinsically and intrinsically. To this end, we develop SimReg, a new debiasing method based on similarity-regularization. SimReg encourages a model to learn unbiased internal representations by either pushing the learned representations towards a model with *good* (unbiased) representations, or pushing it away from a model with biased representations. Our approach is different from previous

[1] Our code is available at: github.com/simreg/SimReg
[2] We use these words interchangeably.
[3] Representation bias is measured by how easy it is to classify whether a given representation stems from a biased sample or not.

Proceedings of the 9th Workshop on Representation Learning for NLP (RepL4NLP-2024), pages 37–50

methods, in that we push the model to learn the "good" behavior from other models, while other approaches usually focus on learning to be different from biased models (Utama et al., 2020a; Sanh et al., 2021; Clark et al., 2019; Nam et al., 2020).

We evaluate our approach on three tasks—natural language inference, fact checking, and paraphrase identification—and multiple spurious correlations attested in the literature: lexical overlap, partial inputs, and unknown biases from weak models (see Section 2.1). We demonstrate that our approach improves performance on out-of-distribution (OOD) challenge sets, while incurring little degradation in in-distribution (ID) performance. Finally, we design an experiment to test the bias remaining in the representations, and find that SimReg models tend to have better performance compared to other debiasing methods.

2 Related Work

A growing body of work has revealed that models tend to exploit spurious correlations found in their training data (Geirhos et al., 2020). Spurious correlations are correlations between certain features of the input and certain labels, which are not causal. Models tend to fail when tested on out-of-distribution data, where said correlations do not hold. We briefly mention several relevant cases and refer to Du et al. (2023) for a recent overview of shortcut learning and its mitigation in natural language understanding.

2.1 Dataset bias

Partial-input bias. A common spurious correlation in sentence-pair classification tasks, like natural language inference (NLI), is **partial-input bias** – the association between words in one of the sentences and certain labels. For example, negation words are correlated with a 'contradiction' label when present in the hypothesis in NLI datasets (Gururangan et al., 2018; Poliak et al., 2018) and with a 'refutes' label when present in the claim in fact verification datasets (Schuster et al., 2019). A common approach for revealing the presence of such spurious correlations is to train a partial-input baseline (Feng et al., 2019). When such a model performs well despite having access only to a part of the input, it indicates that that part has spurious correlations.

Lexical overlap bias. Another common bias is when certain labels are associated with **lexical overlap** between the two input sentences. McCoy et al. (2019) found that high lexical-overlap between the premise and hypothesis correlates with 'entailment' in NLI datasets. As a result, NLI models fail when evaluated on HANS, a challenge set where that correlation does not hold. Similarly, Zhang et al. (2019) found that models trained on a paraphrase identification dataset fail to predict 'non-duplicate' questions that have high lexical-overlap.

Unknown biases. Identifying the preceding biases assumes prior knowledge of the type of bias existing in the dataset. A few studies have used weak learners to identify biases in the dataset without an prior assumption(Sanh et al., 2021; Utama et al., 2020b). Utama et al. (2020b) proposes to train a model on limited number of samples, the hypothesis is that pre-trained models "operate as a rapid surface learners", and will learn the bias in the beginning of the training (i.e., with small number of samples). On the other hand, Sanh et al. (2021) proposed to train a limited capacity models such as Tiny-BERT, where the limited capacity tends the learn and recover previously known biases in the literature.

2.2 Debiasing methods

Spurious correlation mitigation can be performed on different levels: Data-based mitigation, where the data is augmented with samples that do not align with the bias found in the dataset (Wang and Culotta, 2021; Kaushik et al., 2020, inter alia). Model/training-based mitigation, where the either the model or the training procedure is modified. A common strategy in this approach is to train a *bias model*, which latches on the bias in the dataset, and use its outputs to train the final, debiased, *main model*. (He et al., 2019) and (Clark et al., 2019) used variants of product-of-experts (**PoE**) to combine the outputs of the biased and main model during training to encourage the main model to "ignore" biased samples. (Utama et al., 2020a) proposed *confidence regularization* (**ConfReg**), where they perform self-distillation with re-weighted teacher outputs using bias-weighted scaling, i.e., they induce the model to be less confident on biased samples. These methods can be viewed as data re-weighting methods, similar to (Liu et al., 2021), who proposed to up-weigh examples that are miss-classified by the biased model, i.e., hard examples. Similarly, Yaghoobzadeh et al. (2021) proposed to perform additional fine-tuning on *forgettable*

samples after training to increase the robustness ($\mathcal{F}_{\text{BOW}}/\mathcal{F}_{\text{HANS}}$). All these methods work in the output space (extrinsically), while we work in representation space.

Most relevant to our work, Bahng et al. (2020) debias vision models by learning representations that are statistically independent from those of a biased model, by minimizing a statistical independence measure (HSIC) in a min-max optimization objective. We propose a simpler objective function, based on similarity regularization, which can easily be trained by SGD. Additionally, while they focus only on learning representations independent of a biased model, we propose learning representations that are either dissimilar from biased models or similar to unbiased ones.

2.3 Knowledge Distillation

Our approach shares some similarity with Knowledge-Distillation (KD) methods, which transfer knowledge from a teacher model to a (typically smaller) student model. In our framework, we utilize such transfer to improve the robustness of a model. Aguilar et al. (2020) perform KD using internal representations, by minimizing the cosine similarity between the representations of the two models. They compare the similarity of the classification token *(CLS)* whereas we compare all the tokens. Additionally we use second-order isomorphism methods, whereas they use first-order methods.

To our best knowledge, second-order isomorphism methods were previously mainly used for comparing representations and behaviors of models. Our work is one of the first to utilize them to regularize models during training.

3 Methodology

The key idea of our approach is to guide the representation learning of the model in a coarse-grained manner. We achieve this by encouraging the model to learn representations that are either similar to those of an unbiased model or dissimilar from those of a biased model. We design a three-stage procedure (Figure 1):

1. We train a bias model, f_b, on the original training set, $\mathcal{D}$. This model is meant to capture dataset biases, as explained in Section 3.1. In the case of decreasing similarity, we use f_b as our target model, f_g, and continue directly to Stage 3.
2. In order to obtain an unbiased guidance model, we filter the training set based on the predictions of f_b and train a target model f_g on the unbiased part of the training set, $\mathcal{D}^{\mathcal{U}}$ (Section 3.2).
3. We train the main model on $\mathcal{D}$ while encouraging its representations to be (dis)similar to those of f_g (Section 3.3).

3.1 Training a biased model

To mitigate a specific bias (known-bias), we use a bias-specific model, f_b, which is designed to capture that intended bias. For example, to mitigate lexical-overlap bias we use the model proposed in Clark et al. (2019): an MLP whose input features are the ratio of overlap between the two parts of the input, and the average of the minimum cosine similarity between the embeddings of each word from the two sentences. To mitigate unknown-biases, we follow Sanh et al. (2021) and use limited capacity models, such as TinyBert (Turc et al., 2020) and Bag-of-Words (BOW).

In the case of decreasing dissimilarity from a biased model, we use this f_b as the target model, i.e., $f_g = f_b$, and proceed to Stage 3 (Section 3.3). In the case of increasing similarity to an unbiased model, we cannot use f_b as we need an unbiased model; the next section describes how to obtain it.

3.2 Obtaining an unbiased model

To obtain an unbiased guidance model f_g, we run f_b on the training set, $\mathcal{D}$, and exclude samples on which f_b is correct and confident. The remaining samples comprise our unbiased dataset, $\mathcal{D}^{\mathcal{U}}$:

$$\mathcal{D}^{\mathcal{B}} = \{x_i | x_i \in \mathcal{D} \wedge f_b(x) = y_i \wedge c(f_b(x_i)) > c_t\} \quad (1)$$

$$\mathcal{D}^{\mathcal{U}} = \mathcal{D} \setminus \mathcal{D}^{\mathcal{B}} \quad (2)$$

where c_t is a confidence threshold and $c(\cdot)$ is the models' confidence. Our unbiased model, f_g, is obtained by training a new model on $\mathcal{D}^{\mathcal{U}}$.

Choosing the threshold c_t is performed manually by plotting the confidence of the bias model over the training set. When there is a significant bias signal in the dataset, we see a spike in the number of biased samples. Figure 5 (Appendix A.4) shows an example for claim-only bias in FEVER.

A natural question is the following: *What is the advantage of our framework if we already have*

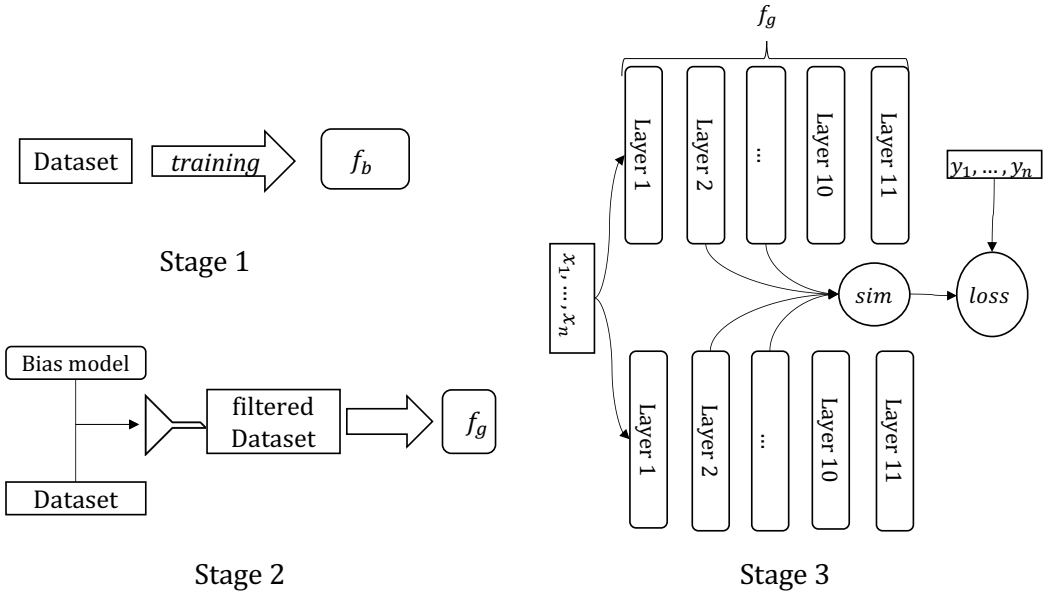

Figure 1: Illustration of SimReg: (1) train a bias model f_b; (2) use its predictions to filter the training set and train a target model f_g; (3) train a main model while guiding its representations to be similar to f_g.

an unbiased model? We emphasize that the unbiased model was trained on $\mathcal{D}^{\mathcal{U}}$, a subset of $\mathcal{D}$, and argue that other samples in $\mathcal{D}$ could also be useful. Indeed, we show experimentally that training a model on the full training set while regularizing it to be similar to the unbiased model leads to a better ID–OOD tradeoff.

3.3 Training the main model

The final step is to train the main model, f_m. We propose two approaches. The first is to encourage the model during training to learn different representations than a biased model, by penalizing its similarity to said biased model, f_b (in this case, $f_g = f_b$). Thus the model would learn different decision boundaries than the biased model. The second approach is to increase the similarity of the learned representations to an unbiased model, f_g. Thus, our model will encode the data in an unbiased manner and its predictions will be less dependant on bias features.

In both cases, we need to compute the similarity between the representations of the main model and those of the target model, f_g. Directly comparing the representations of the models on a single example is not possible, since each model might learn a different latent space for representing the data. Furthermore, the two models might have different architectures and dimensionalities. For instance, in some of our experiments we compare BERT-base (768 dimensions) with TinyBERT (128 dimensions) or with an MLP of 70 dimensions. To overcome these challenges, we use second-order similarity measures, which operate at the batch level (Section 3.3.1).

Formally, we add a similarity regularization term to the batch training loss to promote the similarity/dis-similarity. Given a batch $\mathcal{B}$, we minimize the following objective:

$$\mathcal{L} = \sum_{i \in \mathcal{B}} \mathcal{L}_{CE}(f_m(x_i), y_i) + \lambda \cdot sim(Z, H) \quad (3)$$

where $\mathcal{L}_{CE}$ is the cross-entropy loss, λ is a tradeoff hyper-parameter, Z and H are respectively the main and target model representations of the batch, $f(x)$ is the prediction of the model on input x, and sim is a similarity measure. To increase the similarity, we use $\lambda < 0$.

Since we wish the main model, f_m, to resemble or differ from f_g only on biased samples, we apply regularization only on the biased subset, $\mathcal{D}^{\mathcal{B}}$: We stochastically sample a batch either from $\mathcal{D}^{\mathcal{U}}$ and optimize regular cross-entropy, or from $\mathcal{D}^{\mathcal{B}}$ minimizing the objective in Eq. 3. Section 6.3 shows that regularizing only $\mathcal{D}^{\mathcal{B}}$ results in better OOD performance, supporting our intuition.

3.3.1 Models similarity

Different models may represent the same input differently in their learned latent spaces. Directly comparing vectors from different models can be problematic. To address this, we employ second-order

isomorphism methods. We assess the similarity of the inputs relative to each other within each modality in a given training batch, then compare the similarity matrices of the two modalities to gauge the resemblance of the model encodings. Specifically, we utilize a well-known similarity measure called Centered Kernel Alignment (CKA; Kornblith et al. 2019) with a linear kernel.

4 Experimental Setup

We run our experiments in two settings: (a) Known-bias settings, where we assume the type of bias existing in the dataset, and can construct targeted biased models; and (b) Unknown-bias settings, where the specific type of bias is not presumed, requiring a more general approach of mitigating unknown-biases.

4.1 Datasets

4.1.1 Natural Language Inference

We train models on MNLI, a popular NLI dataset consisting of $\sim 400k$ English examples in multiple genres (Williams et al., 2018). Each example is a pair of premise and hypothesis sentences, and the task is to predict whether the hypothesis is entailed, contradicted, or neutral w.r.t the premise. MNLI contains several spurious correlations as discussed in Section 2, such as lexical overlap and hypothesis-only biases. We train on the MNLI training set and report ID results on dev-matched.

As OOD test set, we use HANS (McCoy et al., 2019) for evaluation against **lexical overlap bias**. HANS is constructed using structured templates that obey bias heuristics, e.g., the hypothesis overlaps with premise, but with half of the examples having non-entailment labels, as opposed to the bias in MNLI. For **hypothesis-only bias** we use MNLI-hard, a subset of MNLI's dev-mismatched set where a hypothesis-only model failed to classify correctly (Gururangan et al., 2018).

4.1.2 Synthetic MNLI

As a sanity test, we introduce synthetic spurious correlations to MNLI (Synthetic-MNLI), following prior work (He et al., 2019; Sanh et al., 2021; Dranker et al., 2021). We prepend the input with a 'label-token' that correlates highly with the label. We used tokens *<0>*, *<1>*, and *<2>*, corresponding to *entailment*, *neutral*, and *contradiction*. Following (Dranker et al., 2021), we denote the probability of injecting a token to the input as the $prevalence$ of the bias, and the probability of the prepended token being correct as the $strength$ of the bias. Through all our experiments, we used prevalence of 1.0 and strength of 0.95. The subsets of examples containing bias token with wrong and right correlations are denoted *anti_bias* and *bias* subsets, respectively.

The goal of this setting is to demonstrate the viability of the proposed approaches. Thus we use an *oracle unbiased* model as f_g for the case of increasing similarity, i.e., a model trained on regular MNLI (without synthetic bias). For the bias model, f_b, we train a model for a small enough number of steps to capture the bias, judging by the rapid drop of training loss; we found 1k steps sufficient.

4.1.3 Fact Verification

Fact Extraction and VERification (FEVER) (Thorne et al., 2018) is a dataset for fact verification against textual sources. Given evidence and claim sentences, the task is to predict the relation between them: SUPPORTED, REFUTED, or NOT ENOUGH INFO. We followed (Schuster et al., 2019) and trained, evaluated on their processed version of FEVER.

Similar to MNLI, the claim part of the input is spuriously correlated with REFUTES label. We use FEVER-Symmetric (Schuster et al., 2019) for OOD evaluation against **claim-only bias**. The construction of FEVER-Symmetric ensures that there is no correlation between partial input and labels, thus it enables us to evaluate the extent of debiasing on this type of bias.

4.1.4 QQP

Quora Question Pairs (QQP) is a collection of >400K question pairs from the Quora platform. Given a pair of questions, the task is to predict whether they are *duplicate* (paraphrase) or *non-duplicate*. QQP is biased in that question pairs with low **lexical-overlap** between them are correlated with the *non-duplicate* label. We train on the QQP training set and evaluate ID on the development set.

Paraphrase Adversaries from Word Scrambling (PAWS) (Zhang et al., 2019) is a dataset for paraphrase identification that is built in a adversarial manner to lexical-overlap bias. The authors scramble the words of a sentence to generate samples with high lexical-overlap that are not a paraphrase. We use the QQP subset of PAWS as our OOD evaluation set for **lexical-overlap bias**.

	MNLI-Hypothesis		MNLI-Lexical		FEVER		QQP	
	dev	MNLI-HARD	dev	HANS	dev	Sym.	dev	PAWS
BERT	83.9	76.9 ± 0.2	84.2	63.6 ± 1.0	85.6	58.4 ± 1.7	91.0	33.3 ± 0.7
f_g	79.1	78.5 ± 0.5	83.0	70.6 ± 0.8	66.8	61.8 ± 0.2	89.1	39.6 ± 0.1
PoE	82.0	$\mathbf{79.5 \pm 0.4}$	83.2	66.6 ± 3.6	78.0	$\mathbf{63.0 \pm 0.6}$	90.5	34.7 ± 0.3
ConfReg	84.3	78.4 ± 0.6	84.3	66.6 ± 3.9	85.2	61.0 ± 1.7	87.4	37.4 ± 1.8
$\mathcal{F}_{HANS}$	-	-	83.9	69.5 ± 0.9	-	-	-	-
SimReg ↑	84.4	79.2 ± 0.3	83.5	$\mathbf{70.5 \pm 1.9}$	80.9	61.6 ± 0.4	89.8	$\mathbf{41.4 \pm 1.2}$
SimReg ↓	83.0	77.9 ± 0.5	84.0	68.5 ± 0.2	84.1	60.3 ± 1.1	90.8	39.0 ± 0.5

Table 1: Known-bias mitigation.

4.2 Models

We evaluate our approach using BERT (Devlin et al., 2018) as both f_g and the main model. We repeat some of the experiments using DeBERTa-V3 (He et al., 2023) to verify that our method is not specific to BERT. For bias modeling, we used an MLP with lexical features as input following Clark et al. (2019) for lexical-bias modeling. For partial-input bias modeling, we simply train BERT with limited input (only on hypothesis / claim for MNLI / FEVER respectively). In unknown-bias modeling, we use TinyBERT (Turc et al., 2020) for MNLI and QQP, and BOW for FEVER, as our limited-capacity model, following Sanh et al. (2021). For full training details, see Appendix A.1.

5 Results

5.1 Synthetic bias

The results on Synthetic-MNLI are in Table 2. All of the SimReg approaches resulted in an increase compared to the baseline on the *anti-biased* subset, where the synthetic token is mis-aligned with the label. Increasing similarity (↑) performed better than decreasing it (↓). The improvement comes at a cost of a small decrease on the biased subset, which is expected. Compared to an oracle model, which was trained without the synthetic bias, the regularized models perform worse, indicating that they were not able to completely discard the bias.

5.2 Known bias

Tables 1 show the results on known bias cases. All our SimReg models outperform the baseline on the OOD test sets. Increasing similarity (↑) seems to work better than decreasing similarity (↓), consistent with synthetic-bias results. In partial-input bias (MNLI-HARD and FEVER), SimReg performs almost as well as PoE on the challenge sets, while PoE has a greater degradation on ID dev sets. Turning to lexical-overlap bias (QQP and MNLI-HANS), we see a similar pattern: SimReg performs much better than the baseline on HANS and PAWS (the OOD sets), with little or no degradation on the corresponding ID dev sets. In contrast, PoE and ConfReg struggle. Generally, increasing similarity works better than decreasing it.

Model	Biased	Anti-biased
BERT-base	98.5 ± 0.1	41.8 ± 1.1
Oracle	83.8	82.1
SimReg ↑	96.7 ± 0.1	61.0 ± 0.9
SimReg ↓	97.0 ± 0.0	49.0 ± 2.4

Table 2: Results on Synthetic-MNLI.

A telling comparison is between SimReg and the guidance model f_g, which is a model that was trained only on unbiased examples (Section 3.2). In most of the cases, when we increase similarity to this model (rows with ↑), we get models that perform better or similar, on both ID and OOD sets. These results support our hypothesis that increasing similarity to an unbiased model can lead to better representations than those of the unbiased model itself by utilizing more data points.

5.3 Unknown bias

The results of unknown bias mitigation are in Table 3. In this settings, we see similar patterns to known-bias results: SimReg outperforms the baseline and the competitive approaches on challenge sets. Interestingly, in these scenarios, the improvement of SimReg over f_g is more prominent, both in challenge datasets and in ID sets.

	MNLI		FEVER		QQP	
	dev	HANS	dev	Symm.	dev	PAWS
BERT	84.2	63.5 ± 1.0	86.0	58.2 ± 0.6	91.1	33.3 ± 0.7
f_g	77.4	64.1 ± 2.2	84.4	61.3 ± 1.0	82.9	48.7 ± 0.9
ConfReg	83.4	63.2 ± 2.1	86.0	60.0 ± 1.6	88.4	32.3 ± 0.4
POE	81.4	68.8 ± 2.0	82.3	61.1 ± 0.8	89.8	40.8 ± 0.1
$\mathcal{F}_{BOW}$	82.8	70.2 ± 1.2	84.0	59.5 ± 2.5	88.1	41.4 ± 5.2
SimReg↑	81.9	$\mathbf{71.4 \pm 0.8}$	84.3	$\mathbf{62.4} \pm 0.6$	84.4	$\mathbf{50.6} \pm 1.9$

Table 3: Unknown-bias mitigation.

5.4 Results with Stronger Models

In this section we investigate whether our approach improves the performance of stronger models than BERT. While most work tends to compare with BERT as the baseline, it is important to demonstrate that a new debiasing method is effective also when applied to stronger models.[4] We experiment with DeBERTa-V3 (He et al., 2023). As Table 4 shows, SimReg still leads to improvements above the strong DeBERTa-V3. where we see similar patterns to the main results, with SimReg↑ outperforming other approaches. Note that we used here the non-entailment subset of HANS to as our OOD evaluation set in MNLI (Lexical-bias and unknown-bias) to emphasize the improvement on the bias-misaligned subset.

6 Analysis

6.1 Similarity Heat-map Analysis

To investigate whether our similarity-based regularization achieves its goal, we compute the similarity between every layer in the main model and every layer in the (unbiased) guidance model, and likewise the similarity between layers of the baseline model and layers of the guidance model. We expect our similarity regularization to increase the similarity of the main model to the guidance model, compared to that of the baseline model.

Figure 2 (Upper) shows that, without similarity-based regularization, the bottom layers of the baseline and guidance model are already similar, but the top layers are rather different. This is consistent with findings on how fine-tuning affects mostly the top layers (Mosbach et al., 2020; Merchant et al., 2020), as both models started from a pre-trained BERT. Figure 2 (Lower) shows that after our similarity-based regularization, the top layers of the main and guidance models become very similar, as desired. Moreover, the regularization also indirectly affects lower layers (bottom row of the heatmap). We conclude that the similarity regularization is successful and affects large parts of the model even when applied only on a few layers.

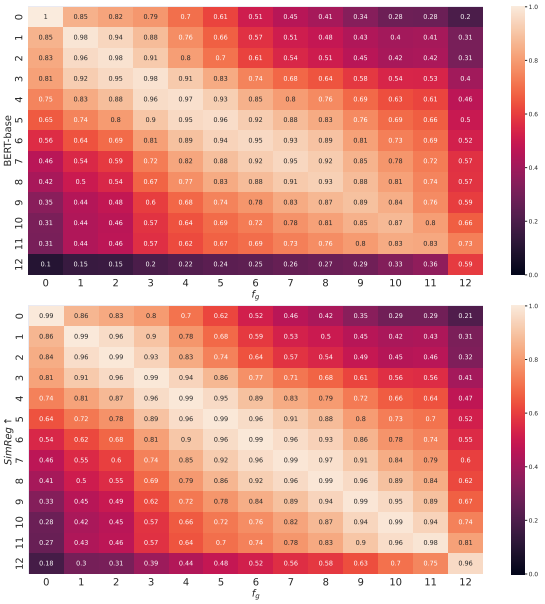

Figure 2: Similarity of an unbiased model, f_g, to either a baseline (Top) or a SimReg model (Bottom). Similarity regularization makes top layers more similar to the unbiased model, as desired.

6.2 Bias recovery

To examine whether representations debiasing does indeed lead to better representations, we designed an experiment to test the bias in the representations. Retraining the classification layer allows us to test to what extent a linear classifier recovers the bias

[4] Bowman (2022) made such a claim about *analyzing* stronger models; we believe it is similarly important to work on *robustifying* stronger models.

	Hypothesis		Lexical	
	ID	HARD	ID	HANS-
baseline	89.9	85.2$_{\pm 0.1}$	89.9	56.7$_{\pm 2.2}$
ConfReg	90.0	86.3$_{\pm 0.2}$	90.3	61.8$_{\pm 1.9}$
SimReg ↑	89.1	**86.5$_{\pm 0.2}$**	89.5	**72.8$_{\pm 0.5}$**
SimReg ↓	89.4	85.8$_{\pm 0.3}$	89.8	63.7$_{\pm 1.5}$

(a) Known-bias (MNLI)

	MNLI		QQP	
	ID	HANS-	ID	PAWS
baseline	89.9	56.7$_{\pm 2.2}$	89.9	55.7$_{\pm 5.6}$
ConfReg	90.1	54.5$_{\pm 2.0}$	88.8	61.1$_{\pm 2.0}$
SimReg ↑	89.1	**66.8$_{\pm 0.7}$**	86.3	**67.0$_{\pm 2.0}$**
SimReg ↓	89.4	61.5$_{\pm 0.2}$	89.2	54.8$_{\pm 1.7}$

(b) Unknown-bias

Table 4: DeBERTa V3 results for MNLI, QQP biases.

existing in the dataset from the representations. In Table 5, we present the results of retraining the classifier of the debiased models in unknown-bias settings. In all approaches we see a drop in OOD accuracy when retraining the classifier [5], consistent with Mendelson and Belinkov (2021)'s observation that debiased models still encode the bias in their representations. However, in SimReg↑ we generally get the highest performance compared to other methods. This indicates that the representations produced by SimReg↑ have the weakest signal of the spurious correlations. We repeated this experiment on debiased models in known-bias settings in App A.5, and found similar patterns.

	MNLI		FEVER	
	dev	HANS-	dev	Sym
BERT	83.9	30.1 $_{\pm 1.3}$	85.4	58.2 $_{\pm 0.1}$
ConfReg	84.8	20.5 $_{\pm 6.2}$	86.0	59.2 $_{\pm 1.0}$
POE	83.0	33.7 $_{\pm 1.6}$	83.4	59.1 $_{\pm 1.5}$
$\mathcal{F}_{BOW}$	83.1	38.1 $_{\pm 0.3}$	84.6	57.0 $_{\pm 1.5}$
SimReg↑	83.9	**41.6 $_{\pm 4.5}$**	85.3	**61.3 $_{\pm 1.3}$**

Table 5: Bias recovery: unknown-bias settings. SimReg↑ shows weakest signal of bias when re-training the classifier.

6.3 Ablations

In this section we perform ablations on SimReg↑ on MNLI datasets in unknown-bias settings (using TinyBERT as f_b). Table 6 shows the results of ablating different parts of our method while keeping the reset unchanged.

SimReg↑$_{BOW}$ refers to When using a bag-of-words model as our limited capacity model f_b, SimReg obtains only slightly worse performance (SimReg↑$_{BOW}$ row). However it also shows that the results can depend on the biases that the weak model f_b discovers.

Using a pre-trained BERT as our guidance model (-f_g row) performs poorly. This highlights that the model that is being used to increase similarity to is an important factor in the process, and that indeed the information is being distilled from f_g into the main model.

The last row shows that applying similarity regularization on the entire training set $\mathcal{D}$ performs poorly. This result supports our intuition in regularizing only the biased samples $\mathcal{D}^{\mathcal{U}}$ (Section 3.3).

ablation	dev.	HANS avg.
SimReg↑	81.9	**71.4**
SimReg↑$_{BOW}$	82.8	70.7
- f_g	82.3	58.1
- Bias regularization	84.2	61.3

Table 6: Ablations on SimReg↑ method.

7 Conclusion

In this work, we have introduced SimReg, a new debiasing approach that employs similarity-based regularization at the representation level. We have demonstrated the effectiveness of SimReg across several NLU tasks, where it notably enhances performance on OOD challenge sets with minimal impact on ID sets.

Additionally, we evaluated the representations of SimReg by testing the amount of bias recovered from the debiased models and found that models debiased using SimReg were least biased after retraining their classifier on a dataset that contains bias. Future work may investigate the effect of simultaneously learning from unbiased and biased models. Another interesting direction is to extend our approach to generation tasks, which would require different similarity measures. Moreover, it is

[5] Check Table 10 in the appendix for HANS- evaluation.

worth testing the efficacy of SimReg on other types of biases such as social biases.

Ethics Statement

Our work develops a new approach to mitigate spurious correlations in NLU tasks. These are also known as dataset biases, but are different from social biases such as gender or racial bias. One could use our approach to debias against social biases. However, a malicious actor could use our basic approach to increase such social bias, rather than decrease it, by reversing the optimization.

Limitations

Similar to most debiasing methods, the success of our method relies on the existing of enough non-biased samples in the training set, which is used to guide our learning process. Additionally, a notable limitation is in the case of debiasing against unknown-bias, where one might speculate that a certain weak model captures the bias, however, it could either miss the bias, or be more powerful and capture additional non-biased samples. In this case, an inspection to the predictions of the weak model might help.

Acknowledgements

This research has been supported by an AI Alignment grant from Open Philanthropy, the Israel Science Foundation (grant No. 448/20), and an Azrieli Foundation Early Career Faculty Fellowship.

References

Qoura question pairs dataset. https://quoradata.quora.com/First-Quora-Dataset-Release-Question-Pairs.

Gustavo Aguilar, Yuan Ling, Yu Zhang, Benjamin Yao, Xing Fan, and Chenlei Guo. 2020. Knowledge distillation from internal representations. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):7350–7357.

Hyojin Bahng, Sanghyuk Chun, Sangdoo Yun, Jaegul Choo, and Seong Joon Oh. 2020. Learning de-biased representations with biased representations.

Samuel Bowman. 2022. The dangers of underclaiming: Reasons for caution when reporting how NLP systems fail. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7484–7499, Dublin, Ireland. Association for Computational Linguistics.

Remi Cadene, Corentin Dancette, Matthieu Cord, Devi Parikh, et al. 2019. Rubi: Reducing unimodal biases for visual question answering. *Advances in Neural Information Processing Systems*, 32:841–852.

Christopher Clark, Mark Yatskar, and Luke Zettlemoyer. 2019. Don't take the easy way out: Ensemble based methods for avoiding known dataset biases. 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 4069–4082, Hong Kong, China. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina N. Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding.

Yana Dranker, He He, and Yonatan Belinkov. 2021. IRM—when it works and when it doesn't: A test case of natural language inference. In *Advances in Neural Information Processing Systems*.

Mengnan Du, Fengxiang He, Na Zou, Dacheng Tao, and Xia Hu. 2023. Shortcut learning of large language models in natural language understanding. *Commun. ACM*, 67(1):110–120.

Shi Feng, Eric Wallace, and Jordan Boyd-Graber. 2019. Misleading failures of partial-input baselines. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5533–5538, Florence, Italy. Association for Computational Linguistics.

R. Geirhos, J.-H. Jacobsen, C. Michaelis, R. Zemel, W. Brendel, M. Bethge, and F. A. Wichmann. 2020. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673.

Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel Bowman, and Noah A. Smith. 2018. 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. Association for Computational Linguistics.

He He, Sheng Zha, and Haohan Wang. 2019. Unlearn dataset bias in natural language inference by fitting the residual. In *Proceedings of the 2nd Workshop on Deep Learning Approaches for Low-Resource NLP (DeepLo 2019)*, pages 132–142, Hong Kong, China. Association for Computational Linguistics.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2023. DeBERTav3: Improving deBERTa using ELECTRA-style pre-training with gradient-disentangled embedding sharing. In *The Eleventh International Conference on Learning Representations*.

Divyansh Kaushik, Eduard Hovy, and Zachary C Lipton. 2020. Learning the difference that makes a difference with counterfactually augmented data. *International Conference on Learning Representations (ICLR)*.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. 2019. Similarity of neural network representations revisited. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 3519–3529. PMLR.

Evan Z Liu, Behzad Haghgoo, Annie S Chen, Aditi Raghunathan, Pang Wei Koh, Shiori Sagawa, Percy Liang, and Chelsea Finn. 2021. Just train twice: Improving group robustness without training group information. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 6781–6792. PMLR.

Tom McCoy, Ellie Pavlick, and Tal Linzen. 2019. 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. Association for Computational Linguistics.

Michael Mendelson and Yonatan Belinkov. 2021. Debiasing methods in natural language understanding make bias more accessible. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1545–1557, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Amil Merchant, Elahe Rahimtoroghi, Ellie Pavlick, and Ian Tenney. 2020. What happens to BERT embeddings during fine-tuning? In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 33–44, Online. Association for Computational Linguistics.

Marius Mosbach, Anna Khokhlova, Michael A. Hedderich, and Dietrich Klakow. 2020. On the interplay between fine-tuning and sentence-level probing for linguistic knowledge in pre-trained transformers. In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 68–82, Online. Association for Computational Linguistics.

Junhyun Nam, Hyuntak Cha, Sungsoo Ahn, Jaeho Lee, and Jinwoo Shin. 2020. Learning from failure: Debiasing classifier from biased classifier. *Advances in Neural Information Processing Systems*, 33:20673–20684.

Hadas Orgad, Seraphina Goldfarb-Tarrant, and Yonatan Belinkov. 2022. How gender debiasing affects internal model representations, and why it matters. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2602–2628, Seattle, United States. Association for Computational Linguistics.

Adam Poliak, Jason Naradowsky, Aparajita Haldar, Rachel Rudinger, and Benjamin Van Durme. 2018. Hypothesis only baselines in natural language inference. In *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics*, pages 180–191, New Orleans, Louisiana. Association for Computational Linguistics.

Victor Sanh, Thomas Wolf, Yonatan Belinkov, and Alexander M Rush. 2021. Learning from others' mistakes: Avoiding dataset biases without modeling them. In *International Conference on Learning Representations*.

Tal Schuster, Darsh Shah, Yun Jie Serene Yeo, Daniel Roberto Filizzola Ortiz, Enrico Santus, and Regina Barzilay. 2019. Towards debiasing fact verification models. 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 3419–3425, Hong Kong, China. Association for Computational Linguistics.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. FEVER: a large-scale dataset for fact extraction and VERification. 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 809–819, New Orleans, Louisiana. Association for Computational Linguistics.

Iulia Turc, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2020. Well-read students learn better: On the importance of pre-training compact models.

Prasetya Ajie Utama, Nafise Sadat Moosavi, and Iryna Gurevych. 2020a. Mind the trade-off: Debiasing NLU models without degrading the in-distribution performance. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8717–8729, Online. Association for Computational Linguistics.

Prasetya Ajie Utama, Nafise Sadat Moosavi, and Iryna Gurevych. 2020b. Towards debiasing NLU models from unknown biases. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7597–7610, Online. Association for Computational Linguistics.

Zhao Wang and Aron Culotta. 2021. Robustness to spurious correlations in text classification via automatically generated counterfactuals. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(16):14024–14031.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. 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.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi 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 Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Yadollah Yaghoobzadeh, Soroush Mehri, Remi Tachet des Combes, T. J. Hazen, and Alessandro Sordoni. 2021. Increasing robustness to spurious correlations using forgettable examples. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 3319–3332, Online. Association for Computational Linguistics.

Yuan Zhang, Jason Baldridge, and Luheng He. 2019. PAWS: Paraphrase adversaries from word scrambling. 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 1298–1308, Minneapolis, Minnesota. Association for Computational Linguistics.

A Appendix

A.1 Training details

We used pre-trained *bert-base-uncased* from HuggingFace models (Wolf et al., 2020) for both the main model training and the guidance model in SimReg↑. Trained for 5 epochs with batch size of 64, for better similarity estimation. For MNLI, QQP we used learning rate of $5e-5$ and $2e-5$ for FEVER, that warms up for 3k steps and decays linearly to 0. The reported results in the tables are the mean and standard deviation of 3 different random seeds. This is true also for competitive methods.

For computing the similarity, we use the mean token representation per layer as the representation of each layer, then we aggregate the similarities of the layers by summing them. We applied the similarity regularization on multiple layers. For increasing similarity we used the last 3 layers from f_g and the main model, following insights from Section A.2.

For decreasing similarity, f_g had a different architecture, we used a combination of layers that ranged across the models. for example, in FEVER claim bias we used first layer, middle layer and last two layers from both f_b and the main model.

As for the threshold c_t, we used 0.8 for unknown-bias experiments, for known bias we used 0.65 except for FEVER claim-bias where we used 0.8. With $\lambda = 100$ for SimReg↑ and $\lambda = 10$ for SimReg↓.

A.2 Layers

In the main experiments, we regularized multiple layers together, as described in Appendix A.1. Our choice of layers is based on Figure 3, where we performed SimReg↑ debiasing on Synthetic-MNLI across layers. The results indicate that deeper layers have the most effect on the debiasing, thus in our main experiments we choose layers 10-12 for regularization in SimReg↑. In decreasing representation similarity, individual layers are not effective, as opposed to regularizing multiple layers as in the main experiments. Thus we chose to regularize in a wide manner over multiple layers.

A.3 Synthetic Bias

In this section we present more detailed results for synthetic-MNLI. In Table 7 we show wider range of configuration for the case of increasing similarity. Note that higher λ values for resulted in models

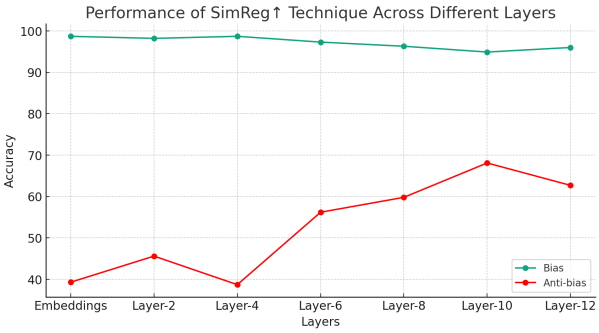

Figure 3: SimReg↑ on Synthetic-MNLI, regularizing one layer at a time.

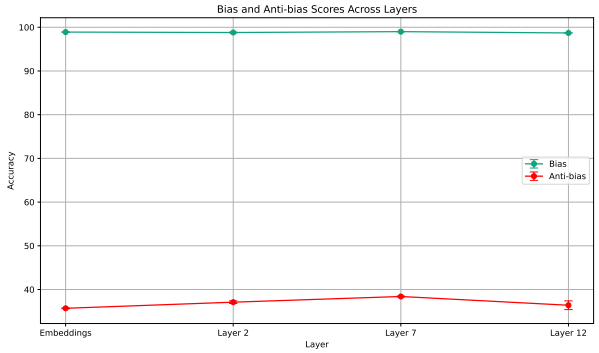

Figure 4: SimReg↓ on Synthetic-MNLI, regularizing one layer at a time.

with better performance on the anti-biased set. Bert-base is BERT trained on Synthetic-MNLI, while BERT (Oracle) is trained on MNLI. In Table 8 we present the case of decreasing similarity. Where we see that in this case, $\lambda = 10$ is a sweet spot, between not changing much ($\lambda = 1$) and changing to much to the level of collapse ($\lambda = 100$).

A.4 Threshold choosing

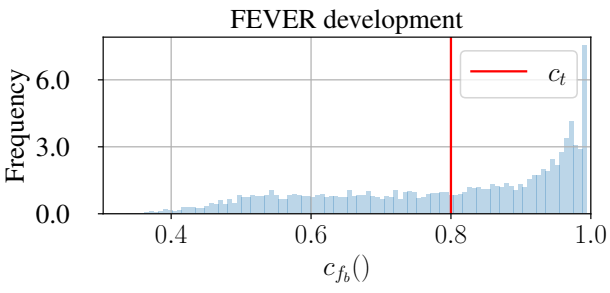

Figure 5: Confidence distribution of a claim-only model (f_b) on FEVER; here $c_t = 0.8$.

A.5 Bias recovery

In Table 9, we present an additional results of our bias-recovery experiments. Where we re-train the

	Biased	Anti-biased	Unbiased
Bert	98.5 ± 0.1	41.8 ± 1.1	78.0 ± 0.4
Oracle	83.5 ± 0.3	82.0 ± 0.9	84.0 ± 0.3
$\lambda = 1$	97.3 ± 0.0	57.7 ± 0.1	83.3 ± 0.2
$\lambda = 10$	96.8 ± 0.1	60.2 ± 0.4	83.6 ± 0.3
$\lambda = 100$	96.7 ± 0.1	61.0 ± 0.9	83.2 ± 0.2

Table 7: SimReg↑: Synthetic-MNLI with prevalence=1 and strength=0.95.

Model	Biased	Anti-biased	Unbiased
Bert-base	98.5 ± 0.1	41.8 ± 1.1	78.0 ± 0.4
f_b	99.9	05.7	64.3
$\lambda = 1$	99.6 ± 0.1	12.2 ± 17.3	53.4 ± 25.6
$\lambda = 10$	96.8 ± 0.2	49.8 ± 2.3	72.4 ± 1.5
$\lambda = 100$	33.5 ± 1.6	32.3 ± 0.4	32.3 ± 1.7

Table 8: SimReg↓: Synthetic-MNLI with prevalence=1 and strength=0.95.

classification layer of the model on the dataset, to test the amount of biased recovered when evaluating the re-trained classifier + model on the OOD challenge sets. Note that on MNLI we evaluate on the bias-misaligned subset of HANS (the non-entailment subset). For results of the models on these subsets before retraining, check Table 10.

We observe that SimReg↑ generally retains high performance on challenge sets after re-training their classifier on the whole dataset $\mathcal{D}$ (with the spurious correlations).

A.6 HANS subsets

Table 10 contains the evaluation of debiasing methods on HANS subsets (non-entailment and entailment).

	IID	HANS -
BERT	83.9 ±0.1	30.1 ±1.3
ConfReg	84.5 ±0.1	7.0 ±2.7
POE	83.6 ±0.1	40.5 ±4.4
SimReg↑	84.0 ±0.1	**42.5 ±1.3**

(a) MNLI Lexical-overlap bias

	dev	FEVER-Sym.
BERT	85.4 ±0.1	58.2 ±0.1
ConfReg	82.1 ±0.4	59.5 ±2.0
POE	79.2 ±0.1	**61.1 ±1.9**
SimReg↑	83.2 ±0.3	60.7 ±0.2

(b) FEVER claim bias

	dev	MNLI-hard
BERT	83.9 ±0.1	76.9 ±0.2
ConfReg	84.5 ±0.2	77.4 ±0.1
POE	83.0 ±0.2	**79.3 ±0.1**
SimReg↑	83.0 ±0.3	79.1 ±0.4

(c) MNLI hypothesis bias

	dev	PAWS
BERT	88.4 ±0.1	28.2 ±2.2
ConfReg	88.0 ±0.1	33.42 ±1.9
POE	90.0 ±0.1	34.1 ±0.1
SimReg↑	90.6 ±0.1	**42.1 ±1.4**

(d) QQP lexical-overlap bias

Table 9: Bias recovery in known-bias settings.

	dev	ent.	non-ent.
BERT	84.2	99.1 ± 0.1	28.1 ± 2.0
f_g	83.0	86.0	55.4
PoE	83.2	77.7 ± 9.8	55.4 ± 7.2
ConfReg	84.3	72.3 ± 8.5	60.9 ± 6.6
$\mathcal{F}_{HANS}$	83.9	–	–
SimReg ↑	83.5	86.4 ± 2.3	54.6 ± 1.6
SimReg ↓	84.0	92.1 ± 0.8	44.8 ± 1.3

(a) Known-bias debiasing.

	dev	ent.	non-ent.
BERT	84.2	99.1 ± 0.1	28.1 ± 2.0
f_g	77.4	53.0 ± 11	75.2 ± 7.4
PoE	81.4	81.1	56.4
ConfReg	83.4	90.0 ± 3.7	36.3 ± 3.7
$\mathcal{F}_{BOW}$	83.0	94.4 ± 1.5	45.9 ± 1.2
SimReg ↑	81.9	78.0 ± 2.5	64.8 ± 1.1
SimReg ↓	82.9	85.6 ± 4.2	41.4 ± 2.5

(b) Unknown-bias debiasing.

Table 10: HANS subsets