# Mitigating Shortcuts in Language Models with Soft Label Encoding

**Zirui He**[1], **Huiqi Deng**[2], **Haiyan Zhao**[1], **Ninghao Liu**[3], **Mengnan Du**[1]
[1]New Jersey Institute of Technology, [2]Shanghai Jiao Tong University, [3]University of Georgia
ziruihe2022@gmail.com, denghq7@sjtu.edu.cn, ninghao.liu@uga.edu
{hz54, mengnan.du}@njit.edu

**Abstract**

Recent research has shown that large language models rely on spurious correlations in the data for natural language understanding (NLU) tasks. In this work, we aim to answer the following research question: *Can we reduce spurious correlations by modifying the ground truth labels of the training data?* Specifically, we propose a simple yet effective debiasing framework, named Soft Label Encoding (SoftLE). First, we train a teacher model to quantify each sample's degree of relying on shortcuts. Then, we encode this shortcut degree into a dummy class and use it to smooth the original ground truth labels, generating soft labels. These soft labels are used to train a more robust student model that reduces spurious correlations between shortcut features and certain classes. Extensive experiments on two NLU benchmark tasks via two language models demonstrate that SoftLE significantly improves out-of-distribution generalization while maintaining satisfactory in-distribution accuracy. Our code is available at https://github.com/ZiruiHE99/sle

**Keywords:** Language models, Robustness, Spurious correlation

## 1. Introduction

Large language models (LLMs), such as BERT (Devlin et al., 2019), RoBERTa (Liu et al., 2019), and GPT-3 (Brown et al., 2020), have achieved remarkable performance in various natural language understanding (NLU) tasks. However, recent studies suggest that these LLMs heavily rely on shortcut learning and spurious correlations rather than developing a deeper understanding of language and semantic reasoning (Mudrakarta et al., 2018; Lapuschkin et al., 2019; Niven and Kao, 2019). Across multiple NLU tasks, this reliance on shortcuts and spurious correlations gives rise to biases within the trained models, which results in their limited generalization capability on out-of-distribution (OOD) datasets (McCoy et al., 2019; Zhang et al., 2019; Yang et al., 2019).

To build more robust models free from biases, several debasing methods have been proposed, e.g., example reweighting that places higher training weights on hard training samples (Schuster et al., 2019), and model ensembling (He et al., 2019; Clark et al., 2019; Mahabadi et al., 2020), which adjust the weights of mislabeled samples to prevent the model from learning spurious associations. Extending these foundational techniques, prevailing methods e.g., confidence regularization (Utama et al., 2020), and LTGR (Du et al., 2021) measure the shortcut degree of each training sample, achieving better results in mitigating shortcut learning. However, the majority of existing debiasing methods rely on manual annotation and necessitate prior knowledge of biased features within the dataset. Manual annotation can be a time-consuming and labor-intensive process, and it remains challenging to comprehensively address bias across the entire dataset. Therefore, the ideal debiasing method should be autonomous and can be widely deployable on different tasks.

Motivated by the crucial observation that the limited robustness of LLMs on NLU tasks arises from spurious correlations learned during training, we aim to improve generalization and robustness by decreasing the likelihood of learning such correlations. Previous work has demonstrated that hard labels cannot express the uncertainty of sample labeling, and some boundary samples are forced to be labeled into definite categories, which increases the risk of overfitting (Xie et al., 2016; Salimans et al., 2016). Recent research on the Natural Language Inference (NLI) task has confirmed similar observations. After many instances are classified into the correct class, their softmax confidences (confidence prediction) will continue to approach the hard labels, contributing to a further reduction in the objective function value (cross-entropy loss) (Tu et al., 2020). This phenomenon is suspected to be detrimental to model generalization, thus we are further motivated to explore the following research question: *Can we reduce spurious correlations by modifying the ground truth labels of the training data?*

In this work, we propose an autonomous debiasing method called Soft Label Encoding (SoftLE) to address the issue of shortcut learning in NLU models through a data-centric perspective. We first train a teacher model with hard labels to determine each sample's degree of relying on shortcuts. We then add one dummy class to encode the shortcut degree, which is used to smooth other dimensions



*LREC-COLING 2024*, pages 11341–11348
20-25 May, 2024. 

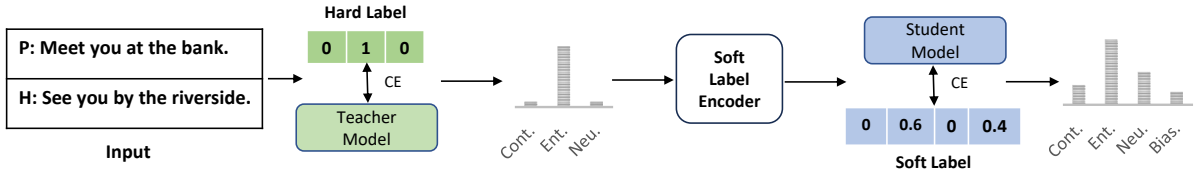


Figure 1: An overview of the proposed Soft Label Encoding framework.

in the ground truth label to generate soft labels (see Figure 1). This new ground truth label is used to train a more robust student model. The key idea of our method is to reduce spurious correlations between shortcut tokens and certain class labels in the training set. This can be leveraged to discourage models from relying on spurious correlations during model training. This also implicitly encourages the models to derive a deeper understanding of the task. This label smoothing method is efficient since it directly operates on labels and does not require manual feature filtering. The major contributions of this work can be summarized as follows:

- We propose a simple yet effective debiasing framework called Soft Label Encoding (SoftLE) to mitigate shortcut learning in natural language understanding models by modifying the ground truth labels during training.
- We provide a theoretical analysis showing how SoftLE reduces spurious correlations between shortcut features and certain class labels in the training data, discouraging models from relying on shortcuts.
- Experimental results demonstrate that SoftLE improves out-of-distribution generalization while maintaining in-distribution accuracy.

## 2. Proposed Method

In this section, we introduce the proposed Soft Label Encoding (SoftLE) debiasing framework.

### 2.1. SoftLE Debiasing Framework

**Problem Formulation.** NLU tasks are usually formulated as a general multi-class classification problem. Consider a dataset $D = \{(x_i, y_i)\}_{i=1}^{N}$ consisting of the input data $x_i \in \mathcal{X}$ and the hard labels $y_i \in \mathcal{Y}$, the goal is to train a robust model with good OOD generalization performance.

**Teacher Model Training.** A biased teacher model $f_T$ containing $K$ classes is first fine-tuned on the corresponding NLU dataset. As shown in Figure 2[1],

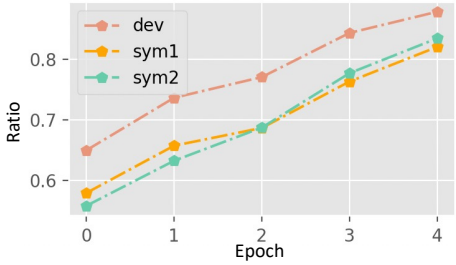


Figure 2: Percentage of over-confident samples in FEVER and OOD test sets during training the teacher model. We set the threshold $\xi$ = 0.9.

when this model starts to converge, the percentage of over-confident samples in the in-domain set exceeds 0.9, while this ratio is around 0.8 for the OOD sets, indicating there are more over-confident samples in in-domain set. The in-domain test set contains both shortcut samples and difficult samples, whereas the two OOD sets primarily contain difficult samples. Therefore, the inconsistency in confidence ratios indicates that samples utilizing more shortcut features will be predicted by the teacher model with a higher softmax confidence. In the following, we leverage the prediction confidence of the model to quantify the degree of shortcut for each training sample.

**Quantifying Shortcut Degree.** We fix the parameters of teacher model $f_T$ and calculate the logit value and softmax confidence of training sample $x_i$ as $z_i^T$ and $\sigma(z_i^T)$ respectively. Then, we set the threshold $\xi$ and hyperparameters to calculate the shortcut degree for each over-confident sample(i.e., $\sigma(z_i^T) > \xi$):

$$s_{i,j} = \log_{\alpha}(\sigma(z_i^T) + \beta). \quad (1)$$

**Soft Label Encoding.** Equipped with the shortcut degree, we then transform a K-class classification problem into a K+1 class problem by introducing a new *dummy category*. The value of the dummy category is given as the shortcut degree value $s_{i,j}$. The original label 1 in one-hot form $y_i$ is transformed into smoothed label: $1 - s_{i,j}$. We illustrate this process using the MNLI task as example (see Figure 1). Here, we set $\xi$ as 0.9. If the teacher model predicts a high softmax confidence $\sigma(z_i^T) > 0.9$ for a sample, then the original three-class classification label $y_i = [0, 1, 0]$ of this sample will be transformed into a four-class label

[1] In Figure 2, 'dev' refers to the development set of the FEVER dataset, and 'sym1' and 'sym2' are the OOD sets of the FEVER dataset.

**Algorithm 1:** Proposed SoftLE framework.

**Input:** Training data $D = \{(x_i, y_i)\}_{i=1}^{N}$.

Set hyperparameters $\alpha$ and $beta$.

**while** *training stage* **do**

- Train teacher network $f_T(x)$. Fix its parameters. Initialize the student network $f_S(x)$;
- Shortcut degree: $s_{i,j} = \log_{\alpha}(\sigma(z_i^T) + \beta)$;
- Put $s_{i,j}$ in the dummy class position;
- Soft label: $1 - s_{i,j}$ for the $1$ position of $y_i$;
- Proposed debiasing training loss: $\mathcal{L}_{SL} = -\sum_{i=1}^{N}\sum_{j=1}^{K+1} y'_{ij}\log(p_{ij})$; Training loss for first few epochs: $\mathcal{L}_{HL} = -\sum_{i=1}^{N}\sum_{j=1}^{K+1} y_{ij}\log(p_{ij})$,;
- Specifically, the first two epochs we use $\mathcal{L}_{HL}$, while later epochs we use $\mathcal{L}_{SL}$.

**while** *inference stage* **do**

- Ignore the dummy class and make the prediction based on values of the first K categories: $\hat{y}_i = \text{argmax}_{j\in[1,\cdots,K]} p_{i,j}$.

$y'_i = [0, 1 - s_{i,j}, 0, s_{i,j}]$. For training samples with a large shortcut degree, more smoothed new labels will be obtained. In contrast, we preserve original hard labels for samples with a low shortcut degree.

### 2.2. Overall Framework

We present the overall framework in Algorithm 1. The dummy class is only required during the training stage and will be discarded during inference.

**The Training Stage.** We use standard cross entropy loss to train the debiased model:

$$\mathcal{L}_{SL} = -\sum_{i=1}^{N}\sum_{j=1}^{K+1} y'_{ij}\log(p_{ij}), \quad (2)$$

where $p_{ij}$ is the predicted probability for input $x_i$ to have label $j$, and $y'_{ij}$ is the *transformed label* for training example $i$.

During training, we replace the proposed loss $L_{SL}$ with the standard hard label loss $L_{HL}$ for the first two epochs as a warming-up training:

$$\mathcal{L}_{HL} = -\sum_{i=1}^{N}\sum_{j=1}^{K+1} y_{ij}\log(p_{ij}), \quad (3)$$

where $y_{ij}$ stands for the *one-hot label* for (K+1)-class classification of the training example. In the last few epochs, we switch back to using $L_{SL}$. This has been demonstrated to retain better ID performance, while achieving similar debiasing performance. We give further analysis in Section 3.3.

**The Inference Stage.** It is worth noting that during the inference stage, we will ignore the dummy class (i.e., the $K+1$ class) and predict based on the first $K$ classes: $\hat{y}_i = \text{argmax}_{j\in[1,\cdots,K]} p_{i,j}$.

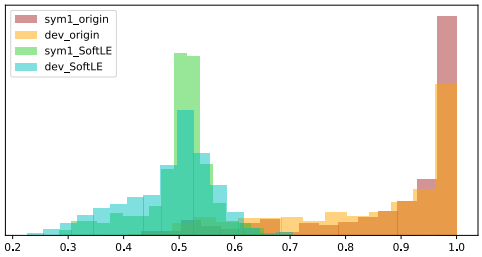


Figure 3: We compared the distribution of softmax confidences for the samples misclassified by softLE and the original model (i.e., teacher model) on Fever and Symm.1. Y-axis denotes the ratio.

## 3. Experiments

In this section, we evaluate the proposed SoftLE debiasing framework to answer three research questions: 1) Compared to baseline methods, can the proposed SoftLE framework achieve the optimal trade-off between in-domain and OOD performance? 2) Why does the proposed SoftLE framework work? 3) What is the key difference between SoftLE and POE in improving the model generalization performance using several benchmark datasets?

### 3.1. Experimental Setting

**Tasks and Datasets** We explore two NLU tasks: natural language inference (NLI) and fact verification. For NLI, we use the MNLI dataset (Williams et al., 2018) to train biased and de-biased models. We evaluate these models on the in-distribution(ID) MNLI-dev set and the out-of-distribution(OOD) HANS dataset (McCoy et al., 2019) to test for generalization. For fact verification, we use the FEVER dataset (Thorne et al., 2018) as our ID data. We then evaluate the model's OOD performance on the FEVER symmetric dataset (Schuster et al., 2019). For both tasks, we employ accuracy as the metric to evaluate the model performance on the ID and OOD sets.Further details are provided in Appendix A.

**Comparing Baselines** We compare our proposed method with four representative baseline methods: *Product of Experts (POE)* (Clark et al., 2019; Mahabadi et al., 2020), *Example Reweighting (ER)* (Schuster et al., 2019), *Regularized Confidence* (Utama et al., 2020), and *Debiasing Masks* (Meissner et al., 2022). More details of the baselines are given in Appendix B.

**Reproducibility** The implementation is based on PyTorch and the Hugging Face package. We aim at complete reproducibility by providing complete code and clear reproduction instructions in our repository.[2]

[2]https://github.com/ZiruiHE99/sle

| Models | MNLI(acc.) | | | FEVER(acc.) | | | |
|---|---|---|---|---|---|---|---|
| | DEV | HANS | Avg. | DEV | Symm.1 | Symm.2 | Avg. |
| Original | **84.5** | 62.4 | 73.5 | 85.6 | 55.1 | 62.2 | 67.6 |
| Reweighting (Schuster et al., 2019) | 81.4 | 68.6 | 75.0 | 84.6 | 61.7 | 64.9 | 70.4 |
| PoE (Clark et al., 2019; Mahabadi et al., 2020) | 84.2 | 64.6 | 74.4 | 82.3 | **62.0** | 64.3 | 69.5 |
| Reg-conf (Utama et al., 2020) | 84.3 | **69.1** | **76.7** | 86.4 | 60.5 | 66.2 | 71.0 |
| Debias-Mask (Meissner et al., 2022) | 81.8 | 68.7 | 75.3 | 84.6 | - | 64.9 | - |
| **SoftLE** | 81.2 | 68.1 | 74.7 | **87.5** | 60.3 | **66.9** | **71.5** |

Table 1: Model performance on in-distribution and OOD test set. We select the version that achieves the best performance in the original paper for the listed baseline methods. The Avg. columns report the average score on in-distribution and challenge sets. We highlight the best performance on each dataset.

### 3.2. Comparison with Baselines

We compare our approach against baselines and report the results in Table 1. We observe that our SoftLE method consistently improves the performance on both challenge sets. However, the in-distribution performance on HANS drops slightly. We also test the framework using RoBERTa-base (Liu et al., 2019). The results are provided in Table 3 for the FEVER task. The results indicate that our proposed method could improve generalization over two challenging OOD test sets while having only a minor sacrifice on the in-domain test set. Figure 3 reveals that despite the biased teacher model assigning high softmax confidences for both in-domain and OOD samples, a larger proportion of high-confidence OOD samples are misclassified. It further illustrates that SoftLE assigns lower softmax confidences for over-confident samples, thereby effectively reducing the probability of the model incorrectly predicting OOD samples.

### 3.3. Ablation Study

In Section 2, we mentioned that a better trade-off between ID and OOD performance can be achieved by adjusting the loss function during training the debiasing model. To confirm that the combination of this adjustment strategy is necessary to achieve our results, we provide an ablation experiment where the debiasing loss $L_{SL}$ is replaced with $L_{HL}$ during different training epochs. Previous work has shown that shortcut features tend to be picked up by the NLU model in very early iterations (Du et al., 2021). Our results on FEVER support this idea, as shown in Table 2, where we find that replacing training loss in the first two epochs outperforms other strategies. As such, SoftLE prevents models from learning spurious correlations, resulting in a lower performance increase on ID and OOD sets during the early stages of training the debiasing model. Thus, this adjustment strategy leverages superficial features, while SoftLE prevents models from solely relying on superficial features, ultimately achieving a delicate balance.

| Method | FEVER | Symm.1 | Symm.2 |
|---|---|---|---|
| Original | 85.6 | 55.1 | 62.2 |
| SoftLE w/o Replacing | 86.6 | 57.7 | 63.9 |
| SoftLE-F2 (Ours) | **87.5** | **60.3** | **66.9** |
| SoftLE-L2 | 87.1 | 57.8 | 64.4 |

Table 2: Our experimental results comparing the original method against several loss function adjustment strategies. SoftLE-F2 denotes training with $L_{HL}$ for the first 2 epochs, while SoftLE-L2 denotes training with $L_{HL}$ for the last 2 epochs.

### 3.4. Why Does Our Algorithm Work?

For an over-confident input sample $x = (x^b, x^{-b})$, let $x^b$ denote *biased features* of the sample, and let $x^{-b}$ represent the remaining features of the sample except for the biased ones. It is generally considered that a bias model only uses the biased features $x^b$ to predict the ground-truth label:

$$p(y^{\text{truth}}|x) = p(y^{\text{truth}}|x^b). \tag{4}$$

Over-confidence indicates that the predicted probability $p(y^{\text{truth}}|x^b)$ of the sample is very high. In other words, for over-confident samples, there is a relatively high spurious correlation between labels and bias features.

In comparison, when we transform the label of the sample, *i.e.*, altering $y^{\text{truth}}$ to $y^{\text{smooth}}$, it is proved in (Clark et al., 2019) that the predictive probability $p(y^{\text{smooth}}|x)$ can be computed as follows:

$$p(y^{\text{smooth}}|x) \propto p(y^{\text{smooth}}|x^{-b})p(y^{\text{smooth}}|x^b). \tag{5}$$

For over-confident samples, we find that the label transformation actually mitigates the potential correlation between labels and biased features. In other words, it significantly lowers the predictive probability given biased features, *i.e.*, $p(y^{\text{smooth}}|x^b) < p(y^{\text{truth}}|x^b)$. Thus, to maximize $p(y^{\text{smooth}}|x)$, our model has to depend more on unbiased features $x^{-b}$ to obtain a higher $p(y^{\text{smooth}}|x^{-b})$ value.

| Method | FEVER | Symm.1 | Symm.2 |
|---|---|---|---|
| Original-RoBERTa | 88.1 | 59.2 | 64.7 |
| SoftLE-RoBERTa | 87.9 | **63.2** | **67.5** |

Table 3: We validated the effectiveness of SoftLE using RoBERTa-base model on FEVER dataset.

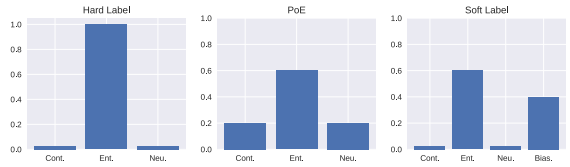


Figure 4: An intuitive comparison between SoftLE framework with original method and Product of Experts (POE). Y-axis denotes softmax confidence.

### 3.5. Difference between SoftLE and POE?

Recent studies indicate that in the PoE training, the debiasing model is encouraged to learn from the errors of the biased model instead of mimicking (Sanh et al., 2021). However, Figure 3 reveals that there exists a gap between misclassified samples of ID and OOD sets. The abuse of spurious correlations results in more misclassified samples with high softmax confidence in the OOD set, and is not consistent with the cause of errors in the ID set. A key difference thus arises: SoftLE method scale down all the over-confident samples while these samples are less focused in PoE training. We also provide an intuitive comparison between the proposed SoftLE framework with standard model training and PoE in Figure 4. The results demonstrate that SoftLE training attains a comparable outcome to the PoE approach concerning confidence reduction. However, the introduction of bias scores reveals its potential to enhance the model's ability to autonomously acquire biases in datasets and focus on the intended NLU tasks.

## 4. Related work

In this section, we briefly review two lines of research that are most relevant to ours.

**Shortcut Learning Phenomenon.** Recent studies indicate that shortcut learning has significantly hurt the models' robustness (Lapuschkin et al., 2019; Niven and Kao, 2019; Du et al., 2023). Fine-tuning pre-trained models can rapidly gain in-distribution improvement while it also gradually increases models' reliance on surface heuristics (Devlin et al., 2019). (He et al., 2019) have demonstrated that a particular label is highly correlated with the presence of several phrases, independent of the other information provided. Several studies have revealed that artificially constructed samples with heuristic features are very likely to trigger erroneous judgments of the model (McCoy et al., 2019; Schuster et al., 2019; Zhang et al., 2019; Yang et al., 2019).

**Shortcut Learning Mitigation.** Previous work demonstrated that training on adversarial data has benefits for generalization capabilities (Wang and Bansal, 2018; Yaghoobzadeh et al., 2021). Several methods were proposed to generate a large-scale data set for the NLI task to reduce annotation artifacts (Zellers et al., 2018; Wang and Culotta, 2020). The main concern is that when generating new data, new biases can be introduced. Some other approaches aim to remove strongly biased samples from the dataset (Sakaguchi et al., 2019; Zhang et al., 2019; Min et al., 2020; Bras et al., 2020). However, these removing-sample methods harm the in-distribution performance significantly, and the criteria for the definition of a bias-free dataset are very obscure (Xiong et al., 2021). Recent studies indicate that a portion of the pre-training model's parameters are correlated with surface statistical patterns (Gordon et al., 2020). Hence pruning this portion of the parameters can reduce the model's memory for spurious correlations (Sanh et al., 2020; Meissner et al., 2022). Most recently, some approaches based on the model interpretability perspective have also been proposed that also have the potential to autonomously identify dataset bias (Wu and Gui, 2022; Wang et al., 2022).

## 5. Conclusions and Future Work

Recently debiasing NLU tasks has attracted increasing attention from the community. We presented SoftLE, an autonomous and efficient framework for debiasing NLU models. By encoding each training sample's degree of relying on shortcuts into soft labels, SoftLE discourages models from learning spurious correlations during training. Across multiple benchmark tasks, SoftLE achieves a favorable trade-off between in-distribution accuracy and out-of-distribution generalization. Our work highlights the promise of data-centric debiasing techniques to build more robust and generalizable language models.

There are various ways to quantify the shortcut degree. Our debiasing framework only attempts one solution to generate the shortcut degree of each sample. Going forward, in order to better measure the shortcut degree of the training samples, a more comprehensive analysis is needed. Additionally, although our proposed debiasing framework is general, we have only applied it to two NLU tasks (MNLI and FEVER) and two types of LLMs (i.e., BERT and RoBERTa). In the future, we plan to extend our debiasing framework to more NLU tasks and additional types of LLMs.

## 6. Bibliographical References


Ronan Le Bras, Swabha Swayamdipta, Chandra Bhagavatula, Rowan Zellers, Matthew E. Peters, Ashish Sabharwal, and Yejin Choi. 2020. Adversarial filters of dataset biases.

Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in Neural Information Processing Systems (NeurIPS)*.

Christopher Clark, Mark Yatskar, and Luke Zettlemoyer. 2019. Don't take the easy way out: Ensemble based methods for avoiding known dataset biases. *Empirical Methods in Natural Language Processing (EMNLP)*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. *North American Chapter of the Association for Computational Linguistics (NAACL)*.

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

Mengnan Du, Varun Manjunatha, Rajiv Jain, Ruchi Deshpande, Franck Dernoncourt, Jiuxiang Gu, Tong Sun, and Xia Hu. 2021. Towards interpreting and mitigating shortcut learning behavior of nlu models. *North American Chapter of the Association for Computational Linguistics (NAACL)*.

Mitchell A. Gordon, Kevin Duh, and Nicholas Andrews. 2020. Compressing bert: Studying the effects of weight pruning on transfer learning.

He He, Sheng Zha, and Haohan Wang. 2019. Unlearn dataset bias in natural language inference by fitting the residual. *2019 EMNLP workshop*.

Sebastian Lapuschkin, Stephan Wäldchen, Alexander Binder, Grégoire Montavon, Wojciech Samek, and Klaus-Robert Müller. 2019. Unmasking clever hans predictors and assessing what machines really learn. *Nature Communications*, 10(1).

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Rabeeh Karimi Mahabadi, Yonatan Belinkov, and James Henderson. 2020. End-to-end bias mitigation by modelling biases in corpora. In *58th Annual Meeting of the Association for Computational Linguistics (ACL)*.

R Thomas McCoy, Ellie Pavlick, and Tal Linzen. 2019. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. *57th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Johannes Mario Meissner, Saku Sugawara, and Akiko Aizawa. 2022. Debiasing masks: A new framework for shortcut mitigation in nlu. In *Conference on Empirical Methods in Natural Language Processing*.

Junghyun Min, R. Thomas McCoy, Dipanjan Das, Emily Pitler, and Tal Linzen. 2020. Syntactic data augmentation increases robustness to inference heuristics. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*.

Pramod Kaushik Mudrakarta, Ankur Taly, Mukund Sundararajan, and Kedar Dhamdhere. 2018. Did the model understand the question? *56th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Timothy Niven and Hung-Yu Kao. 2019. Probing neural network comprehension of natural language arguments. *57th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2019. Winogrande: An adversarial winograd schema challenge at scale.

Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. 2016. Improved techniques for training gans.

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

Victor Sanh, Thomas Wolf, and Alexander M Rush. 2020. Movement pruning: Adaptive sparsity by fine-tuning. *34th Conference on Neural Information Processing Systems (NeurIPS)*.

Tal Schuster, Darsh J Shah, Yun Jie Serene Yeo, Daniel Filizzola, Enrico Santus, and Regina Barzilay. 2019. Towards debiasing fact verification models. *Empirical Methods in Natural Language Processing (EMNLP)*.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. Fever: a large-scale dataset for fact extraction and verification. *North American Chapter of the Association for Computational Linguistics (NAACL)*.

Lifu Tu, Garima Lalwani, Spandana Gella, and He He. 2020. An empirical study on robustness to spurious correlations using pre-trained language models. *Transactions of the Association for Computational Linguistics (TACL)*.

Prasetya Ajie Utama, Nafise Sadat Moosavi, and Iryna Gurevych. 2020. Mind the trade-off: Debiasing nlu models without degrading the in-distribution performance. *58th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Tianlu Wang, Rohit Sridhar, Diyi Yang, and Xuezhi Wang. 2022. Identifying and mitigating spurious correlations for improving robustness in nlp models.

Yicheng Wang and Mohit Bansal. 2018. Robust machine comprehension models via adversarial training.

Zhao Wang and Aron Culotta. 2020. Robustness to spurious correlations in text classification via automatically generated counterfactuals.

Adina Williams, Nikita Nangia, and Samuel R Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. *North American Chapter of the Association for Computational Linguistics (NAACL)*.

Ting Wu and Tao Gui. 2022. Less is better: Recovering intended-feature subspace to robustify NLU models. In *Proceedings of the 29th International Conference on Computational Linguistics*.

Lingxi Xie, Jingdong Wang, Zhen Wei, Meng Wang, and Qi Tian. 2016. Disturblabel: Regularizing cnn on the loss layer.

Ruibin Xiong, Yimeng Chen, Liang Pang, Xueqi Cheng, Zhi-Ming Ma, and Yanyan Lan. 2021. Uncertainty calibration for ensemble-based debiasing methods. In *Advances in Neural Information Processing Systems*, volume 34, pages 13657–13669. Curran Associates, Inc.

Yadollah Yaghoobzadeh, Soroush Mehri, Remi Tachet, T. J. Hazen, and Alessandro Sordoni. 2021. Increasing robustness to spurious correlations using forgettable examples.

Yinfei Yang, Yuan Zhang, Chris Tar, and Jason Baldridge. 2019. PAWS-X: A Cross-lingual Adversarial Dataset for Paraphrase Identification. In *Empirical Methods in Natural Language Processing (EMNLP)*.

Rowan Zellers, Yonatan Bisk, Roy Schwartz, and Yejin Choi. 2018. Swag: A large-scale adversarial dataset for grounded commonsense inference. *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Yuan Zhang, Jason Baldridge, and Luheng He. 2019. Paws: Paraphrase adversaries from word scrambling. *arXiv preprint arXiv:1904.01130*.

## A. Details of Tasks and Datasets

**Natural Language Inference (NLI)** is a task that involves understanding and inferring logical relationships between linguistic texts. We select the MNLI dataset (Williams et al., 2018) to train biased and de-biased models that will evaluate in-distribution performance on the MNLI-dev dataset and out-of-distribution performance on the HANS dataset (McCoy et al., 2019). The samples in the HANS dataset are generated by some simple heuristic rules that enable correct classification by relying only on surface features (e.g., word overlap, negation, etc.). The purpose of the HANS dataset is to assess how well the model really performs in terms of inference ability, rather than relying only on shallow surface features.

**Fact Verification** is a task that evaluates computer models to make inferences and judgments about the accuracy of a given factual statement. The FEVER dataset (Thorne et al., 2018) provides claim-evidence pairs and labels for three categories: Supports, Refutes, and NEI(Not Enough Info). The FEVER symmetric dataset (Schuster et al., 2019) contains 2 subsets with 717 and 712 manually generated claim-evidence pairs, respectively, where the synthetic pairs hold the same relationships (e.g., SUPPORTS or REFUTES) but express different and opposite facts. The goal is to verify whether relying on the cues of the claims leads to incorrect predictions.

## B. Details of Baselines

**Product of Experts (POE)** (Clark et al., 2019; Mahabadi et al., 2020) is ensemble learning-based technique where the predictions of multiple "expert" models are combined by taking their product.

**Example Reweighting (ER)** (Schuster et al., 2019) allocates greater weights to instances of the minority class, consequently incentivizing the model to give increased attention to these instances, thereby enhancing its capacity to accurately identify the less represented class.

**Regularized Confidence** (Utama et al., 2020) is motivated by the fact that overconfidence can indicate that the model is not well-calibrated and may perform poorly on unseen data. To address this issue, a regularization term is added to the loss function to encourage the model to output a more uniform (or less confident) probability distribution.

**Debiasing Masks** (Meissner et al., 2022) removes specific weights of the network that is associated with biased behavior without altering the original model. A mask search is performed to identify and remove those weights.