Improving the robustness of NLI models with minimax training

Michalis Korakakis
University of Cambridge
mk2008@cam.ac.uk

Andreas Vlachos
University of Cambridge
av308@cam.ac.uk

Abstract

Natural language inference (NLI) models are susceptible to learning shortcuts, i.e. decision rules that spuriously correlate with the label. As a result, they achieve high in-distribution performance, but fail to generalize to out-of-distribution samples where such correlations do not hold. In this paper, we present a training method to reduce the reliance of NLI models on shortcuts and improve their out-of-distribution performance without assuming prior knowledge of the shortcuts being targeted. To this end, we propose a minimax objective between a learner model being trained for the NLI task, and an auxiliary model aiming to maximize the learner's loss by up-weighting examples from regions of the input space where the learner incurs high losses. This process incentivizes the learner to focus on under-represented "hard" examples with patterns that contradict the shortcuts learned from the prevailing "easy" examples. Experimental results on three NLI datasets demonstrate that our method consistently outperforms other robustness enhancing techniques on out-of-distribution adversarial test sets, while maintaining high in-distribution accuracy.

1 Introduction

Natural language inference (NLI)[1] models have achieved state-of-the-art results on many benchmarks (Wang et al., 2019). However, recent work has demonstrated that their success is partly due to learning and using shortcuts (Gururangan et al., 2018; Poliak et al., 2018; Geirhos et al., 2020), i.e. spurious correlations between input attributes and labels introduced during dataset creation.[2] For example, high word-overlap between the premise and the hypothesis in the MNLI (Williams et al., 2018) dataset is strongly correlated with the entailment label (McCoy et al., 2019). Consequently, NLI models that exploit shortcuts perform well on in-distribution samples, but are brittle when tested on out-of-distribution adversarial test sets that explicitly target such phenomena (Naik et al., 2018; Glockner et al., 2018).

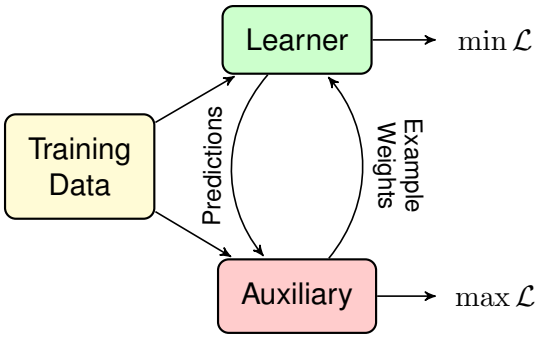


Figure 1: Illustration of the proposed minimax training objective. The learner optimizes for the NLI task, whereas the auxiliary tries to maximize the learner's loss by generating an example weight distribution that encourages the learner to focus on "hard" examples with patterns that contradict the shortcuts. At inference time the learner can make predictions without the auxiliary.

Thus, numerous methods have been proposed to prevent models from learning shortcuts present in NLI datasets (Belinkov et al., 2019; Schuster et al., 2019; Zhou and Bansal, 2020; Stacey et al., 2020; Du et al., 2021; Modarressi et al., 2023, *inter alia*). Most approaches typically assume access to an auxiliary model designed to rely on shortcuts for predictions. The output of the auxiliary is then used to re-weight training instances for the learner model via ensembling (He et al., 2019; Clark et al., 2019; Karimi Mahabadi et al., 2020). However, knowing the shortcuts in advance assumes domain- and dataset-specific knowledge, which is not always available and limits the potential of shortcut mitigation (Rajaee et al., 2022).

A separate line of work overcomes this issue by

[1] Also known as textual entailment (Dagan et al., 2005).

[2] In this paper, we use the terminology introduced by Geirhos et al. (2020). Other works also refer to shortcuts as "biases" and/or "heuristics."

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 14322–14339
July 9-14, 2023

forcing the auxiliary model to learn and exploit shortcuts either by exposing it to only a small number of training examples (Utama et al., 2020b), or by leveraging an auxiliary with reduced learning capabilities (Clark et al., 2020a; Sanh et al., 2021). Another approach is to fine-tune an already trained NLI model on examples that were frequently misclassified during the initial training stage (Yaghoobzadeh et al., 2021). While these works show promising results, they assume that the learner will naturally exploit the same types of shortcuts as the auxiliary. In practise, the behavior of the learner diverges from that of the auxiliary. For instance, Amirkhani and Pilehvar (2021) empirically demonstrate that commonly used auxiliaries often down-weight examples that are useful for training the learner, while Xiong et al. (2021) show that inaccurate uncertainty estimations by the auxiliary model can hinder the learner's out-of-distribution generalization capabilities.

In this paper, we propose a training method to reduce the reliance of NLI models on shortcuts in order to improve their out-of-distribution performance. To this end, we frame training as a minimax objective between a learner and an auxiliary (Figure 1). The learner optimizes for the NLI task, whereas the auxiliary tries to maximize the loss of the learner by up-weighting "hard" examples. The key insight behind our training method is that NLI models suffer from poor performance on under-represented "hard" training instances with patterns that contradict the shortcuts found in the dominant "easy" examples (Tu et al., 2020). Therefore, by encouraging the learner to perform well on these examples and rely less on the "easy" examples with shortcuts, we can obtain better out-of-distribution generalization.

Compared to existing robustness enhancing techniques, our training method (i) does not assume knowledge of the shortcuts being targeted, (ii) detects and up-weights examples in a data-driven way, i.e. the auxiliary is a parameterized neural network that predicts weights for each training instance at every training iteration, and (iii) uses a small feed-forward network rather than a large-scale pre–trained language model (PLM) for the auxiliary, thus incurring a small computational overhead.

We evaluate our proposed method in three commonly-used NLI datasets, namely, MNLI (Williams et al., 2018), FEVER (Thorne et al., 2018), and QQP (Iyer et al., 2017), and their corresponding out-of-distribution adversarial test sets, HANS (McCoy et al., 2019), Symmetric (Schuster et al., 2019), and PAWS (Zhang et al., 2019). We observe that compared to other state-of-the-art robustness enhancing methods, the minimax training objective consistently improves out-of-distribution performance. We further verify the effectiveness of the minimax objective using a synthetic shortcut experimental setup, and then show that the performance gains generalize to a range of large-scale PLMs, out-of-domain test sets, and a question answering dataset. Finally, we empirically analyze the minimax objective to obtain further insights in the workings of the proposed training method.

2 Minimax Training for Shortcut Mitigation

Suppose we have a dataset $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$ comprising the input data $x_i \in \mathcal{X}$ and the labels $y_i \in \mathcal{Y}$. Our goal is to learn a model $f_\theta : \mathcal{X} \rightarrow \mathcal{Y}$ parameterized by θ. The standard training objective for achieving this is empirical risk minimization (ERM) that minimizes the average training loss:

$$J(\theta) = \min_{\theta} \frac{1}{N} \sum_{i=1}^{N} \ell(f_\theta(x_i), y_i), \qquad (1)$$

where $\ell(f_\theta(x_i), y_i)$ is the cross-entropy loss. When shortcuts are present in the "easy" examples that are well-represented in the training data, ERM-trained models will exploit them to achieve low training loss. Consequently, this will lead to poor performance on under-represented "hard" examples where such shortcuts do not hold. These examples are pivotal for ensuring good generalization performance on out-of-distribution samples (Yaghoobzadeh et al., 2021). Crucially, the loss of "hard" examples decreases considerably more slowly than the average loss throughout training (Tu et al., 2020). Therefore, our aim is to obtain a weight distribution that places emphasis on the under-represented "hard" examples, where we minimize the weighted training loss:

$$J(\theta) = \min_{\theta} \sum_{i=1}^{N} w_i \ell(f_\theta(x_i), y_i), \qquad (2)$$

where w_i is the weight associated with the i-th example x_i. Intuitively, the example weights should

Algorithm 1: Minimax Training.

Input: Dataset $\mathcal{D}$, learner f_θ, auxiliary g_ϕ, mini-batch size n, # of iterations T
Output: optimized learner f_θ
pre-train f_θ on $\mathcal{D}$ using ERM
for $\tau = 1, \ldots, T$ **do**
  sample mini-batch $\{x_i, y_i\}_{i=1}^n$ from $\mathcal{D}$
  generate weights via $g_{\phi(x_i, y_i)}$
  generate predictions via $f_\theta(x_i, y_i)$
  update θ to
  $$\min \sum_{i=1}^{n} g_\phi(x_i, y_i)\ell(f_\theta(x_i), y_i)$$
  update ϕ to
  $$\max \sum_{i=1}^{n} g_\phi(x_i, y_i)\ell(f_\theta(x_i), y_i)$$
end

have high values for the under-represented "hard" instances, and low values for the prevailing "easy" instances with shortcuts.

To compute the example weight distribution, we propose a minimax training objective between a learner f_θ and an auxiliary $g_\phi : \mathcal{X} \times \mathcal{Y} \rightarrow [0, 1]$ parameterized by ϕ. Both models are optimized in an alternating fashion. The learner f_θ tries to minimize the loss for the classification task (NLI in this paper). The task of the auxiliary g_ϕ is to maximize the learner's loss by generating a weight for each training example at every training iteration, such that the learner is incentivized to concentrate on regions of the input space where it incurs high losses. Thus, the learner will prioritize learning from under-represented "hard" examples that counteract the use of shortcuts present in the dominant "easy" examples. Formally, the minimax objective can be written as:

$$J(\theta, \phi) = \min_\theta \max_\phi \sum_{i=1}^{N} g_\phi(x_i, y_i)\ell(f_\theta(x_i), y_i). \quad (3)$$

Both θ and ϕ can be optimized using any standard optimization algorithm, such as stochastic gradient descent. In order to ensure that the example weights lie in the range $[0, 1]$, the output of the auxiliary model is passed through a sigmoid function. At test time the learner can make predictions without relying on the auxiliary. Algorithm 1 summarizes the overall training procedure.

3 Experimental Setup

3.1 Data

We conduct experiments using three English NLI datasets, MNLI, FEVER, and QQP. For each dataset, we evaluate performance on an out-of-distribution adversarial test set constructed to examine model reliance on specific shortcuts for predictions.

MNLI The MNLI (Williams et al., 2018) dataset contains approximately 430k premise-hypothesis pairs labelled as entailment if the premise entails the hypothesis, contradiction if it contradicts the hypothesis, or neutral otherwise. We evaluate in-distribution performance on MNLI-matched and out-of-distribution performance on HANS (McCoy et al., 2019), an adversarial test set designed to investigate whether a model exploits the high-word overlap shortcut to predict entailment.

FEVER We conduct experiments on FEVER (Thorne et al., 2018), a fact verification dataset containing around 180k pairs of claim–evidence pairs. The goal in FEVER is to predict whether the retrieved evidence supports a claim, refutes a claim, or there is not enough information. As we are interested in the NLI part of the task, we assume the gold evidence given. We further evaluate on Symmetric (Schuster et al., 2019), which is designed to address the claim-only shortcut, whereby a model learns to use only the claim for predictions while ignoring the evidence.

QQP The QQP (Iyer et al., 2017) dataset contains over 400k question pairs annotated as either paraphrase or non-paraphrase. We evaluate out-of-distribution performance on PAWS (Zhang et al., 2019), an adversarial test set constructed to penalize the high-word overlap shortcut that models exploit to predict the paraphrase label.

3.2 Models

Following previous work (Sanh et al., 2021; Utama et al., 2020b; Yaghoobzadeh et al., 2021), we use BERT (Devlin et al., 2019) and conduct experiments with BERT-base as the learner model. We use a 3-layer multiple-layer perceptron (MLP) for the auxiliary with $\tanh$ as the activation function for the middle layer. Furthermore, we normalize the weights of the auxiliary to have a mean weight of 1 across the batch, and add a constant value to every example weight to ensure that all

Method	MNLI		FEVER		QQP	
	Dev	HANS	Dev	Sym.	Dev	PAWS
ERM	**84.4**	62.6	85.7	55.1	**90.8**	36.0
Shortcut is known in advance						
PoE (Karimi Mahabadi et al., 2020)†	84.2	66.3	84.4	66.2	-	-
Regularized-conf (Utama et al., 2020a)†	84.3	69.1	86.4	60.5	89.1	40.0
No prior shortcut knowledge						
PoE + CE (Sanh et al., 2021)†	83.2	67.9	85.3	57.9	-	-
Self-debias (Utama et al., 2020b)†	82.3	69.7	-	-	85.2	**57.2**
$\mathcal{F}_{\text{BOW}}$ (Yaghoobzadeh et al., 2021)†	83.1	70.5	**87.1**	61.0	89.0	48.8
Minimax (Ours)	83.6±0.2	**72.8**±0.4	85.4±0.6	**62.5**±0.3	87.9±0.5	53.7±1.8

Table 1: Accuracies on the MNLI, FEVER, and QQP datasets, along with their corresponding adversarial test sets, HANS, Symmetric (Sym.), and PAWS. Numbers are averaged on 5 runs with standard deviations. † are reported results and underlying indicates statistical significance against the ERM-trained BERT-base baseline.

examples contribute to the loss in order to avoid filtering useful "easy" examples and hurting in-distribution performance, i.e. $w_i + c > 0$, with $c = 1$. We obtain word representations for the auxiliary using 300-dimensional Glove (Pennington et al., 2014) embeddings and averaging them. We use Adam (Kingma and Ba, 2015) to train both the auxiliary and the learner model. For the learner model we use default architectures and hyperparameters from the Hugging Face Transformers library (Wolf et al., 2020). Finally, we pre-train the learner model for 3 epochs to ensure that it will initially prioritize learning the shortcuts. We train models for 10 epochs and report the mean and standard deviation over 5 runs with different random seeds. Finally, we conduct statistical significance using a two-tailed t-test (with $p < 0.05$).

3.3 Baselines

We compare our method with representative techniques from two robustness enhancing categories. The first category assumes that the shortcut being targeted for mitigation is known in advance. We use the method of Karimi Mahabadi et al. (2020) (PoE) which ensembles the auxiliary and the learner via the product-of-experts (Hinton, 2002), so that the learner will focus on examples that the auxiliary cannot predict well. We also consider confidence regularization (Utama et al., 2020a) (Regularized-conf) which relies on a self-knowledge distillation objective to down-weight examples for which the auxiliary provides over-confident predictions.

The second robustness enhancing category includes approaches that do not assume any prior shortcut knowledge. We use the method of Utama et al. (2020b) (Self-debias), who propose to exploit a "shallow" model trained on a small fraction of the training data as the auxiliary model. Sanh et al. (2021) (PoE + CE) use BERT-tiny (Turc et al., 2019) as a "weak" (low-capacity) auxiliary model, and train it on the full training dataset. Finally, Yaghoobzadeh et al. (2021) ($\mathcal{F}_{\text{BOW}}$) first train the model on the entire dataset, and then fine-tune it on the "forgettable examples" (Toneva et al., 2019), i.e. samples that during the initial training stage were either classified correctly and misclassified afterwards, or they were never properly classified.

4 Results

Main Results Table 1 presents the main experimental results. In general, we see that in all settings the minimax objective significantly improves out-of-distribution performance on the adversarial test sets compared to the ERM-trained BERT-base baseline. In particular, it outperforms the latter on HANS, Symmetric, and PAWS by 10.2, 7.4, and 17.7, respectively. However, we also observe that training using the minimax objective results in a small reduction in the in-distribution performance. Specifically, on MNLI, FEVER, and QQP, the decrease in the in-distribution accuracy is 0.8, 0.3, and 2.9, respectively. Compared to other state-of-the-art robustness enhancing techniques, our method improves in-distribution accuracy on MNLI and out-of-distribution performance on HANS and Symmetric. Conversely, on QQP, $\mathcal{F}_{\text{BOW}}$ and Reguralized-conf outperform minimax

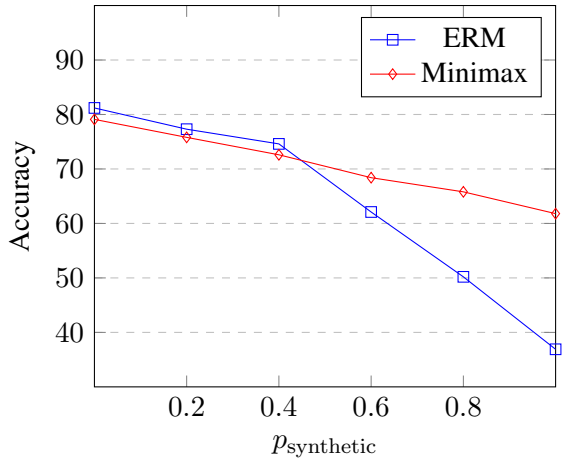


Figure 2: Accuracies for models trained on a modified version of MNLI with a synthetic shortcut, i.e. a prefix in the hypothesis containing the ground-truth label with probability $p_{\text{synthetic}}$. On the MNLI test set the prefix is always a random label.

training by 1.1 and 1.2, while on PAWS Self-debias improves out-of-distribution performance by 3.5. Notably, the improvement for Self-debias comes at the expense of a considerable drop in the in-distribution performance on QQP, i.e. 5.6 reduction in accuracy for Self-debias compared to the ERM-trained BERT-base model.

Synthetic Shortcut Following previous work (He et al., 2019; Clark et al., 2019; Sanh et al., 2021), we modify the MNLI training data by adding a synthetic shortcut, i.e. a prefix in the hypothesis containing the ground-truth label with probability $p_{\text{synthetic}}$ or alternatively a random label. Conversely, on the modified MNLI test set the prefix is always a random label. If the model exploits the synthetic shortcut for predictions, then it will have low accuracy on the test set. Figure 2 shows that the performance of the ERM-trained BERT-base model deteriorates rapidly on the test set as the number of training examples containing the synthetic shortcut increases, whereas the reduction in performance for the model trained with the minimax objective is much less drastic.

Out-of-Domain Generalization We further investigate the generalization capabilities of models trained using the proposed minimax objective on various out-of-domain NLI datasets. To this end, following the experimental setup of Karimi Mahabadi et al. (2020) and Sanh et al. (2021), we train models on SNLI (Bowman et al., 2015), and evaluate performance on AddOneRTE (AdOne) (Pavlick and Callison-Burch, 2016), Definite Pronoun Resolution (DPR) (Rahman and Ng, 2012), Semantic Proto-Roles (SPR) (Reisinger et al., 2015), FrameNet+ (FN+) (Pavlick et al., 2015), SciTail (Khot et al., 2018), GLUE diagnostic test set (Wang et al., 2018), and the SNLI-hard test set (Gururangan et al., 2018). Table 2 presents the results. In general, we observe that our method consistenly outperforms the ERM-trained BERT-base baseline, with the only exception being the GLUE diagnostic test set, where the latter improves accuracy by 0.24. Furthermore, we see that the minimax training outperforms PoE and PoE + CE in five out of 7 out-of-domain test sets.

Domains	ERM	PoE	PoE + CE	Minimax
ADD1	86.54	**87.42**	87.20	87.25
DPR	49.92	49.85	50.10	**50.16**
SPR	58.71	61.58	60.99	**61.86**
FN+	53.98	54.01	54.18	**54.23**
SCITAIL	70.14	71.32	73.75	**75.19**
GLUE	55.62	**55.93**	54.82	55.38
SNLI-hard	81.07	81.39	81.62	**81.81**

Table 2: Accuracies on various out-of-domain test sets for a BERT-base model trained on SNLI with empirical risk minimization (ERM), PoE (Karimi Mahabadi et al., 2020), PoE + CE (Sanh et al., 2021), and the proposed minimax objective.

Model	OOD Test Set	ERM	Minimax
BERT-large	HANS	71.6	**77.3**
	Symmetric	60.1	**69.2**
	PAWS	38.6	**57.8**
RoBERTa-large	HANS	74.9	**79.1**
	Symmetric	67.6	**73.6**
	PAWS	38.8	**56.6**
XLNet-large	HANS	76.1	**78.6**
	Symmetric	68.5	**76.2**
	PAWS	44.7	**62.9**

Table 3: Accuracies on HANS, Symmetric, and PAWS for large-scale pre-trained language models trained with empirical risk minimization (ERM), and our proposed minimax training objective.

Large-scale Pre-trained Language Models We examine whether the performance improvements of training the BERT-base model using the minimax objective also transfer to large-scale PLMs. In particular, we conduct experiments with BERT-large (Devlin et al., 2019) (340M parameters), RoBERTa-large (Liu et al., 2019) (340M parame-

Method	SQuAD	AddSent	AddOneSent
ERM	**88.72**	54.10	58.96
PoE + CE	86.49	56.80	61.04
Minimax	86.51	**57.36**	**62.13**

Table 4: F1 scores for a BERT-base model trained on SQuAD with empirical risk minimization (ERM), PoE + CE (Sanh et al., 2021), and the proposed minimax objective.

ters), and XLNet-large (Yang et al., 2019) (355M parameters). The experimental results in Table 3 demonstrate that our method yields substantial performance gains over ERM for all three large-scale PLMs. In particular, on HANS, Symmetric, and PAWS, minimax training improves performance compared to ERM for BERT-large by 5.7, 9.1, and 19.2, for RoBERTa-large by 4.2, 6, and 17.8, and finally, for XLNet-large by 2.5, 7.7, and 18.2, respectively.

Question Answering Following Sanh et al. (2021), we also conduct experiments on a question answering dataset. In particular, we train BERT-base models on SQuAD (Rajpurkar et al., 2016), and evaluate their out-of-distribution performance on the Adversarial SQuAD dataset (Jia and Liang, 2017). Table 4 shows that minimax improves out-of-distribution performance on the AddSent and AddOneSent adversarial test sets compared to the ERM-trained BERT-base baseline and PoE + CE.

5 Analysis

Using the loss to detect "hard" examples We investigate whether the loss provides a robust signal for discovering "hard" examples that contradict the shortcuts found in the "easy" examples. To this end, we manually classify training instances from MNLI into two categories, namely, "easy" entailment instances with a large amount of words occurring both in the hypothesis and the premise, and under-represented "hard" non-entailment examples with high word overlap, and study their average losses during training. Figure 3 demonstrates that the high-loss examples on MNLI are dominated by the "hard" non-entailment category, whereas the "easy" entailment examples incur predominantly low-losses.

Removing "easy" examples and inverting the weight distribution We evaluate whether we can improve the overall performance of the minimax

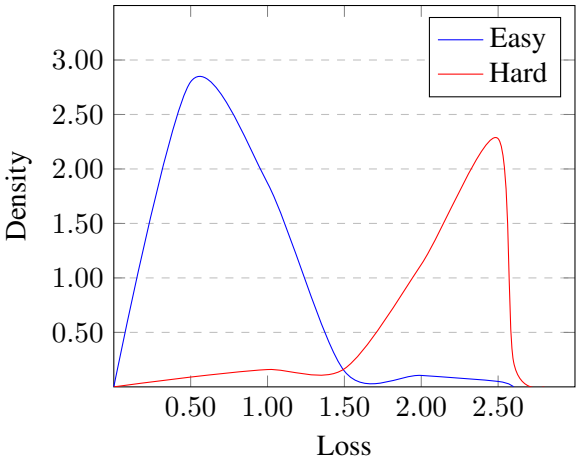


Figure 3: Histogram of average losses on MNLI for "easy" examples with shortcuts, i.e. entailment instances with a large amount of words occurring both in the hypothesis and the premise, and "hard" examples with patterns that contradict them, i.e. non-entailment examples with high word overlap.

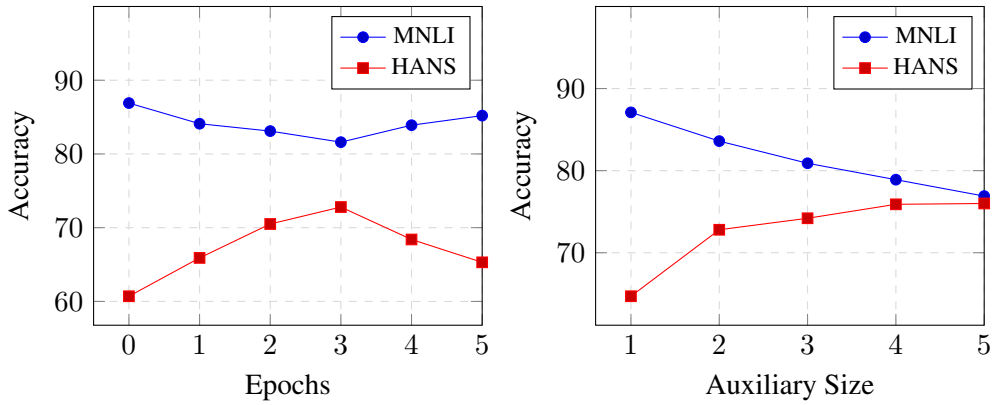
objective by discarding the "easy" examples (filtering minimax), i.e. removing their contribution to the loss by allowing the auxiliary to generate example weights $w_i \geq 0$ via setting $c = 0$. Furthermore, we also examine whether the learnt example weight distribution is meaningful, by keeping the order of the examples fixed and inverting their weights (inverse minimax), i.e. the examples with the largest weights get the lowest weights and vice versa. The experimental results in Table 5 show that using filtering minimax results in similar out-of-distribution performance to that of standard minimax, however, the drop in the in-distribution performance for the former is much more considerable. Conversely, the inverse minimax objective leads to high in-distribution accuracy (similar to that of ERM-trained models), at the expense of out-of-distribution performance.

Effect of number of epochs for pre-training the learner We investigate how performance changes as we vary the number of epochs required for pre-training the learner model. Figure 4a demonstrates that out-of-distribution performance is high when we pre-train the learner for 2 and 3 epochs, but drops when the duration of the pre-training stage is either too short or too long, which consequently results in less informative losses for the auxiliary to learn from.

Impact of size of the auxiliary We explore whether the size of the auxiliary impacts the

Method	MNLI		FEVER		QQP	
	Dev	HANS	Dev	Sym.	Dev	PAWS
ERM	**84.4**	62.6	**85.7**	55.1	**90.8**	36.0
Minimax	83.6	**72.8**	85.4	**62.5**	87.9	**53.7**
Filtering Minimax	80.1	69.9	81.7	61.7	83.8	51.3
Inverse Minimax	84.3	59.6	85.5	53.2	90.6	31.2

Table 5: Accuracies on the MNLI, FEVER, and QQP datasets, along with their corresponding adversarial test sets, HANS, Symmetric (Sym.), and PAWS, for models trained with ERM, minimax, filtering minimax, and inverse minimax. The minimax and ERM rows repeat results from Table 1.



(a) Number of epochs for pre-training the learner. (b) Different auxiliary sizes (hidden layers).

Figure 4: Effect of different components on the learner's in-distribution (MNLI) and out-of-distribution performance (HANS).

learner's in-distribution and out-of-distribution performance. To this end, we train the learner using several auxiliary models of varying sizes. Specifically, we make auxiliary models larger by increasing the number of hidden layers while keeping the other hyperparameters constant. We observe that varying the capacity of the auxiliary model affects the learner's in-distribution and out-of-distribution performance (Figure 4b). In particular, the out-of-distribution performance of the learner model increases as the auxiliary model becomes stronger up to a certain point, while in-distribution performance drops slightly at first and then more strongly. Finally, we observe that increasing the size of the auxiliary has the side effect of incentivizing it to learn the trivial solution of maximising all example weights.

Examining the weighted examples We use the converged auxiliary model to present examples of down-weighted and up-weighted training instances on MNLI. Table 6 demonstrates that the auxiliary is able to correctly identify, and subsequently, down-weight "easy" examples, i.e. entailment with a large amount of words occurring in the premise and hypothesis, and up-weight "hard" examples with patterns that directly contradict the shortcut, i.e. non-entailment with high word overlap. Furthermore, Figure 5 visualises the distribution of the MNLI example weights learned at the end of training. We observe that the minimax objective does not use the trivial solution of setting all weights to 1 to maximize the learner's loss. Conversely, the example weights form two main clusters, at both ends of the histogram.

6 Related Work

Distributionally Robust Optimization Training objectives for ensuring strong model performance across all samples typically fall under the framework of distributionally robust optimization (DRO) (Ben-Tal et al., 2013; Duchi and Namkoong, 2018). DRO seeks to minimize the worst-case loss by placing emphasis on "hard" examples. Sagawa et al. (2020) extend the DRO framework to the case where the training data belongs to predefined groups (e.g. demographic

	Label	Premise	Hypothesis	Example Weight
Down-weighted	Entailment	The doctor was paid by the actor.	The doctor paid the actor.	1.13
	Entailment	The doctors visited the lawyer.	The lawyer visited the doctors.	1.16
	Entailment	The secretaries encouraged the scientists and the actors.	The secretaries encouraged the actors.	1.25
	Entailment	The athlete who the judges admired called the manager.	The judges admired the athlete.	1.29
Up-weighted	Contradiction	A subcategory of accuracy is consistency.	Accuracy is a subcategory of consistency.	3.34
	Contradiction	Of course, we never rejected people for being too flaky.	We rejected people for being flaky.	3.36
	Contradiction	A subcategory of accuracy is consistency.	Accuracy is a subcategory of consistency.	3.41
	Neutral	Some people do I know.	I do know some people.	3.65
	Neutral	Grace and consistency?	Consistency?	3.69

Table 6: Weighted examples by the auxiliary model at the end of training on MNLI.

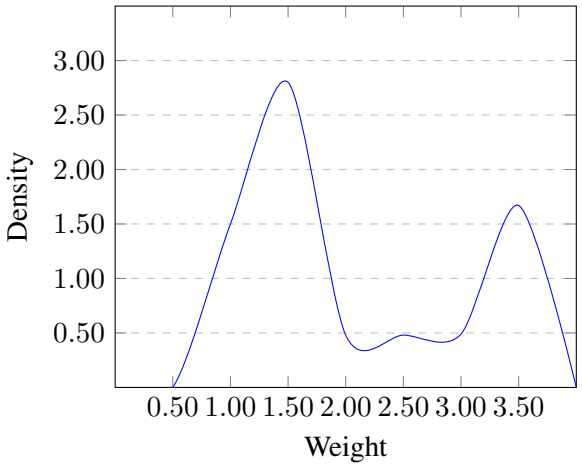


Figure 5: Histogram of example weights generated by the auxiliary model on MNLI at the end of the training.

groups), and then focus on improving the worst-group performance. Our proposed method is closest to research works that assume that group annotations are not available during training. For instance, Bao et al. (2021) develop group-specific classifiers to discover groupings, Sohoni et al. (2020) cluster the data, and Liu et al. (2021) propose a two-stage approach, by first training a model for a few epochs, and then using a second model to up-weight examples that the first model misclassified. However, these approaches require access to a validation set with group annotations, and/or rely on fixed example weights to determine groupings, i.e. they do not dynamically monitor the learner's training dynamics.

Example Weighting Our proposed training objective is also related to example weighting methods that are typically used to mitigate dataset-related issues, such as label noise and class imbalance. For example, approaches like focal loss (Lin et al., 2017) encourage the model to focus on "hard" instances. Conversely, in self-paced learning (Kumar et al., 2010), the example weights emphasize training using the "easy" examples first. Recent works focus on learning a weighting scheme with gradient-based (Fan et al., 2018; Raghu et al., 2021; Wang et al., 2020) and meta-learning methods (Jiang et al., 2018; Ren et al., 2018; Shu et al., 2019). While our proposed method also learns example weights in a data-driven way, we do so using a minimax training objective that places emphasis on up-weighting examples with patterns that contradict the shortcuts. Finally, Zhou et al. (2022) present a related shortcut-agnostic mitigation method by framing the task of generating an example weight distribution as nested optimization, where in the lower-level objective the model minimizes the weighted ERM loss, and on the upper-level objective the example weights are updated to minimize an out-distribution criterion. Our approach is different since we incorporate two models into the training process, while Zhou et al. (2022) use the same model in both the lower- and the upper-level objectives.

Dataset Filtering Another line of related work focuses on improving robustness by filtering the training data instead of modifying the training objective and/or the model architecture. Zellers et al. (2018) and Zellers et al. (2019) use this ap-

proach to mitigate the inclusion of shortcuts during dataset creation. Sakaguchi et al. (2020) propose AFLITE, an adversarial method for filtering examples with shortcuts. AFLITE works by training several models over small partitions of the initial dataset to discover "easy" examples that contain shortcuts. Wu et al. (2022) fine-tune a PLM to synthetically generate NLI examples, and then use z-statistics (Gardner et al., 2021) to remove samples with shortcuts.However, dataset filtering methods may hinder in-distribution performance, due to removing useful examples that contribute towards learning the underlying task. Conversely, our proposed minimax training objective assigns low weights to "easy" examples instead of completely eliminating them, thus preserving in-distribution performance.

Generative Adversarial Networks The minimax objective we propose is reminiscent of the training objective of generative adversarial networks (GANs) (Goodfellow et al., 2014). In NLP, GANs are commonly used to address exposure bias in text generation (de Masson d'Autume et al., 2019). However, in practise, they perform worse than simpler methods (Caccia et al., 2020). A separate family of methods focuses on using the training objective of GANS to improve the computational efficiency of language modelling pretraining (Clark et al., 2020b). Closer to our work, adversarial training (Miyato et al., 2017) aims to improve robustness in text classification, but this method only operates at the level of word embeddings used in representing a single sentence, and thus is not applicable to NLI.

7 Conclusion

In this work, we present a minimax training objective for reducing the reliance of NLI models on shortcuts in order to improve their overall robustness without assuming prior knowledge about the existence of specific shortcuts. Our proposed method leverages an auxiliary model that tries to maximize the learner's loss by up-weighting underrepresented "hard" examples with patterns that contradict the shortcuts present in the prevailing "easy" examples. Experiments across three NLI datasets demonstrate that our minimax objective consistently improves performance on various out-of-distribution adversarial test sets.

Limitations

Since the minimax objective requires using two separately trained models, i.e. the learner and the auxiliary, the design of the latter plays a crucial role in the overall stability of the training process. In particular, while having a very capable auxiliary model will naturally result in a more accurate and robust example weight distribution, it will also potentially lead to overfitting to certain training instances with high-losses. Another potential limitation of minimax training is that the existence of noise in the labels may cause the auxiliary to generate erroneous example weights due to high-loss noisy instances co-existing with the "hard" examples containing meaningful patterns that contradict the shortcuts. Furthermore, we explore shortcut mitigation only for NLI in English, and thus our method might not transfer to other tasks and/or languages. Finally, the datasets we consider are well-used and -discussed in the literature, and consequently their shortcuts (and how they are adopted by the models) are well-known. Further testing is needed to establish whether our approach would transfer to datasets containing different shortcuts.

Acknowledgements

The authors wish to thank Pasquale Minervini and Tal Schuster for their helpful comments and feedback. Michalis Korakakis is supported by the Cambridge Commonwealth, European and International Trust and the ESRC Doctoral Training Partnership. Andreas Vlachos is supported by the ERC grant AVeriTeC (GA 865958).

References

Hossein Amirkhani and Mohammad Taher Pilehvar. 2021. Don't discard all the biased instances: Investigating a core assumption in dataset bias mitigation techniques. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4720–4728, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Yujia Bao, Shiyu Chang, and Regina Barzilay. 2021. Predict then interpolate: A simple algorithm to learn stable classifiers. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 640–650. PMLR.

Yonatan Belinkov, Adam Poliak, Stuart Shieber, Benjamin Van Durme, and Alexander Rush. 2019. Don't

take the premise for granted: Mitigating artifacts in natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 877–891, Florence, Italy. Association for Computational Linguistics.

Aharon Ben-Tal, Dick den Hertog, Anja De Waegenaere, Bertrand Melenberg, and Gijs Rennen. 2013. Robust solutions of optimization problems affected by uncertain probabilities. *Manag. Sci.*, 59(2):341–357.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal. Association for Computational Linguistics.

Massimo Caccia, Lucas Caccia, William Fedus, Hugo Larochelle, Joelle Pineau, and Laurent Charlin. 2020. Language gans falling short. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

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.

Christopher Clark, Mark Yatskar, and Luke Zettlemoyer. 2020a. Learning to model and ignore dataset bias with mixed capacity ensembles. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3031–3045, Online. Association for Computational Linguistics.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020b. ELECTRA: pre-training text encoders as discriminators rather than generators. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Ido Dagan, Oren Glickman, and Bernardo Magnini. 2005. The PASCAL recognising textual entailment challenge. In *Machine Learning Challenges, Evaluating Predictive Uncertainty, Visual Object Classification and Recognizing Textual Entailment, First PASCAL Machine Learning Challenges Workshop, MLCW 2005, Southampton, UK, April 11-13, 2005, Revised Selected Papers*, volume 3944 of *Lecture Notes in Computer Science*, pages 177–190. Springer.

Cyprien de Masson d'Autume, Shakir Mohamed, Mihaela Rosca, and Jack W. Rae. 2019. Training language gans from scratch. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 4302–4313.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. 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. Association for Computational Linguistics.

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. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 915–929, Online. Association for Computational Linguistics.

John C. Duchi and Hongseok Namkoong. 2018. Learning models with uniform performance via distributionally robust optimization. *CoRR*, abs/1810.08750.

Yang Fan, Fei Tian, Tao Qin, Xiang-Yang Li, and Tie-Yan Liu. 2018. Learning to teach. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

Matt Gardner, William Merrill, Jesse Dodge, Matthew Peters, Alexis Ross, Sameer Singh, and Noah A. Smith. 2021. Competency problems: On finding and removing artifacts in language data. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1801–1813, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard S. Zemel, Wieland Brendel, Matthias Bethge, and Felix A. Wichmann. 2020. Shortcut learning in deep neural networks. *Nat. Mach. Intell.*, 2(11):665–673.

Max Glockner, Vered Shwartz, and Yoav Goldberg. 2018. Breaking NLI systems with sentences that require simple lexical inferences. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 650–655, Melbourne, Australia. Association for Computational Linguistics.

Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville, and Yoshua Bengio. 2014. Generative adversarial networks. *CoRR*, abs/1406.2661.

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.

Geoffrey E. Hinton. 2002. Training products of experts by minimizing contrastive divergence. *Neural Comput.*, 14(8):1771–1800.

S. Iyer, N. Dandekar, and K. Csernai. 2017. First quora dataset release: Question pairs. *Accessed online at https://www.quora.com/q/quoradata/First-Quora-Dataset-Release-Question-Pairs*.

Robin Jia and Percy Liang. 2017. Adversarial examples for evaluating reading comprehension systems. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2021–2031, Copenhagen, Denmark. Association for Computational Linguistics.

Lu Jiang, Zhengyuan Zhou, Thomas Leung, Li-Jia Li, and Li Fei-Fei. 2018. Mentornet: Learning data-driven curriculum for very deep neural networks on corrupted labels. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 2309–2318. PMLR.

Rabeeh Karimi Mahabadi, Yonatan Belinkov, and James Henderson. 2020. End-to-end bias mitigation by modelling biases in corpora. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8706–8716, Online. Association for Computational Linguistics.

Tushar Khot, Ashish Sabharwal, and Peter Clark. 2018. Scitail: A textual entailment dataset from science question answering. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 5189–5197. AAAI Press.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

M. Pawan Kumar, Benjamin Packer, and Daphne Koller. 2010. Self-paced learning for latent variable models. In *Advances in Neural Information Processing Systems 23: 24th Annual Conference on Neural Information Processing Systems 2010. Proceedings of a meeting held 6-9 December 2010, Vancouver, British Columbia, Canada*, pages 1189–1197. Curran Associates, Inc.

Tsung-Yi Lin, Priya Goyal, Ross B. Girshick, Kaiming He, and Piotr Dollár. 2017. Focal loss for dense object detection. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 2999–3007. IEEE Computer Society.

Evan Zheran 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, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 6781–6792. PMLR.

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. *CoRR*, abs/1907.11692.

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.

Takeru Miyato, Andrew M. Dai, and Ian J. Goodfellow. 2017. Adversarial training methods for semi-supervised text classification. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Ali Modarressi, Hossein Amirkhani, and Mohammad Taher Pilehvar. 2023. Guide the learner: Controlling product of experts debiasing method based on token attribution similarities. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1954–1959, Dubrovnik, Croatia. Association for Computational Linguistics.

Aakanksha Naik, Abhilasha Ravichander, Norman Sadeh, Carolyn Rose, and Graham Neubig. 2018. Stress test evaluation for natural language inference. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2340–2353, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Ellie Pavlick and Chris Callison-Burch. 2016. Most "babies" are "little" and most "problems" are "huge": Compositional entailment in adjective-nouns. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2164–2173, Berlin, Germany. Association for Computational Linguistics.

Ellie Pavlick, Travis Wolfe, Pushpendre Rastogi, Chris Callison-Burch, Mark Dredze, and Benjamin Van Durme. 2015. FrameNet+: Fast paraphrastic tripling of FrameNet. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 408–413, Beijing, China. Association for Computational Linguistics.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. 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.

Aniruddh Raghu, Maithra Raghu, Simon Kornblith, David Duvenaud, and Geoffrey E. Hinton. 2021. Teaching with commentaries. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Altaf Rahman and Vincent Ng. 2012. Resolving complex cases of definite pronouns: The Winograd schema challenge. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 777–789, Jeju Island, Korea. Association for Computational Linguistics.

Sara Rajaee, Yadollah Yaghoobzadeh, and Mohammad Taher Pilehvar. 2022. Looking at the overlooked: An analysis on the word-overlap bias in natural language inference. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10605–10616, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Drew Reisinger, Rachel Rudinger, Francis Ferraro, Craig Harman, Kyle Rawlins, and Benjamin Van Durme. 2015. Semantic proto-roles. *Transactions of the Association for Computational Linguistics*, 3:475–488.

Mengye Ren, Wenyuan Zeng, Bin Yang, and Raquel Urtasun. 2018. Learning to reweight examples for robust deep learning. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 4331–4340. PMLR.

Shiori Sagawa, Pang Wei Koh, Tatsunori B. Hashimoto, and Percy Liang. 2020. Distributionally robust neural networks. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2020. Winogrande: An adversarial winograd schema challenge at scale. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 8732–8740. AAAI Press.

Victor Sanh, Thomas Wolf, Yonatan Belinkov, and Alexander M. Rush. 2021. Learning from others' mistakes: Avoiding dataset biases without modeling them. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

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.

Jun Shu, Qi Xie, Lixuan Yi, Qian Zhao, Sanping Zhou, Zongben Xu, and Deyu Meng. 2019. Meta-weight-net: Learning an explicit mapping for sample weighting. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 1917–1928.

Nimit Sharad Sohoni, Jared Dunnmon, Geoffrey Angus, Albert Gu, and Christopher Ré. 2020. No subclass left behind: Fine-grained robustness in coarse-grained classification problems. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Joe Stacey, Pasquale Minervini, Haim Dubossarsky, Sebastian Riedel, and Tim Rocktäschel. 2020. Avoiding the Hypothesis-Only Bias in Natural Language Inference via Ensemble Adversarial Training. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8281–8291, Online. 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.

Mariya Toneva, Alessandro Sordoni, Remi Tachet des Combes, Adam Trischler, Yoshua Bengio, and Geoffrey J. Gordon. 2019. An empirical study of example forgetting during deep neural network learning. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

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*, 8:621–633.

Iulia Turc, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Well-read students learn better: The impact of student initialization on knowledge distillation. *CoRR*, abs/1908.08962.

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.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019. Superglue: A stickier benchmark for general-purpose language understanding systems. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 3261–3275.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

Xinyi Wang, Hieu Pham, Paul Michel, Antonios Anastasopoulos, Jaime G. Carbonell, and Graham Neubig. 2020. Optimizing data usage via differentiable rewards. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 9983–9995. PMLR.

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, New Orleans, Louisiana. 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.

Yuxiang Wu, Matt Gardner, Pontus Stenetorp, and Pradeep Dasigi. 2022. Generating data to mitigate spurious correlations in natural language inference datasets. *CoRR*, abs/2203.12942.

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 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 13657–13669.

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.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime G. Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 5754–5764.

Rowan Zellers, Yonatan Bisk, Roy Schwartz, and Yejin Choi. 2018. SWAG: A large-scale adversarial dataset for grounded commonsense inference. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 93–104, Brussels, Belgium. Association for Computational Linguistics.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4791–4800, Florence, Italy. 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.

Xiang Zhou and Mohit Bansal. 2020. Towards robustifying NLI models against lexical dataset biases. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8759–8771, Online. Association for Computational Linguistics.

Xiao Zhou, Yong Lin, Renjie Pi, Weizhong Zhang, Renzhe Xu, Peng Cui, and Tong Zhang. 2022. Model agnostic sample reweighting for out-of-distribution learning. In *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pages 27203–27221. PMLR.

A Training Details

In this section, we detail the models and hyperparameters we use in our experiments. For all experiments, the auxiliary model is optimized using the Adam optimizer $\beta = (0.9, 0.999), \epsilon = 1\text{e}{-8}$, with a learning rate of $1\text{e}{-3}$. We use the HuggingFace implementation of `BERT-base-uncased` as our learner model.

MNLI We use the following hyper-parameters for the learner model: a learning rate of $5\text{e}{-5}$ and a batch size of 32. The learning rate is linearly increased for 2000 warming steps and linearly decreased to 0 afterward. We use an Adam optimizer $\beta = (0.9, 0.999), \epsilon = 1\text{e}{-8}$, and add a weight decay of 0.1.

FEVER We use the following hyper-parameters for the learner model: a learning rate of $2\text{e}{-5}$, and a batch size of 32. The learning rate is linearly increased for 1500 warming steps and linearly decreased to 0 afterward. We use an Adam optimizer $\beta = (0.9, 0.999), \epsilon = 1\text{e}{-8}$, and add a weight decay of 0.1.

QQP We use the following hyper-parameters for the learner model: a learning rate of $5\text{e}{-5}$, and a batch size of 32. The learning rate is linearly increased for 1000 warming steps and linearly decreased to 0 afterward. We use an Adam optimizer $\beta = (0.9, 0.999), \epsilon = 1\text{e}{-8}$, and add a weight decay of 0.1.

B Additional Experimental Results

Domains	ERM	Minimax	Inv. Minimax
ADD1	86.54	**87.25**	57.48
DPR	49.92	**50.16**	38.13
SPR	58.71	**61.86**	39.63
FN+	53.98	**54.23**	37.52
SCITAIL	70.14	**75.19**	54.68
GLUE	**55.62**	55.38	41.74
SNLI-hard	81.07	**81.81**	59.40

Table 7: Accuracies on various out-of-domain test sets for a BERT-base model trained on SNLI with empirical risk minimization (ERM), the proposed minimax objective, and inverse minimax. The ERM and Minimax columns repeat results from Table 2.

Inverse Minimax - Out of Domain Generalization We train the inverse minimax model (which is incentivized to up-weight the "easy" examples with shortcuts) on SNLI, and evaluate performance on several out-of-distribution test sets. From the results in Table 7 we observe that the out-of-distribution performance of the inverse minimax model is considerably worse compared to the ERM-trained baseline and the model trained using the proposed minimax objective.

Additional Results for MNLI In Table 8, we show the performance of our method on MNLI-matched (MNLI-m) and MNLI-mismatched (MNLI-mm), and their corresponding hard sets.

Additional Results for HANS Table 9 shows detailed accuracy scores on the three shortcut categories of HANS. Overall, compared to the ERM-trained BERT-base model minimax training retains satisfactory performance in the entailment class, and provides considerable improvements for non-entailment. Specifically, on the Lexical Overlap, Constituent, and Subsequence shortcut categories, the decrease in accuracy in entailment for minimax training compared to the ERM-trained BERT-base model is 7.1, 1.6, and 2.5, while for non-entailment performance improves by 20.2, 9.4, and 36.7, respectively.

Method	MNLI-m		MNLI-mm		MNLI-m hard		MNLI-mm hard	
	Dev	Test	Dev	Test	Dev	Test	Dev	Test
ERM	**84.4**	84.1	83.9	83.1	76.1	75.2	77.4	75.5
PoE (Karimi Mahabadi et al., 2020)†	84.2	84.1	**84.8**	83.4	**78.0**	76.8	79.2	76.8
Regularized-conf (Utama et al., 2020a)†	84.3	84.1	85.0	84.2	-	78.3	-	77.3
PoE + CE (Sanh et al., 2021)†	83.2	-	83.5	-	-	77.6	-	76.3
Minimax (Ours)	83.6	**84.8**	83.6	**85.6**	77.9	**79.4**	**79.9**	**78.7**

Table 8: Accuracies on MNLI-matched (MNLI-m), MNLI-mismatched (MNLI-mm), MNLI-matched hard, and MNLI-mismatched hard. † are reported results.

Method	Lexical Overlap		Constituent		Subsequence	
	Entailment	Non-Entailment	Entailment	Non-Entailment	Entailment	Non-Entailment
ERM	**98.9**	51.2	**99.3**	10.8	**99.4**	15.7
Minimax	91.8	**71.4**	97.7	**20.2**	96.9	**52.4**

Table 9: Accuracies on the HANS Lexical Overlap, Constituent, and Subsequence shortcut categories.

ACL 2023 Responsible NLP Checklist

A For every submission:

☐ A1. Did you describe the limitations of your work?
Left blank.

☐ A2. Did you discuss any potential risks of your work?
Left blank.

☐ A3. Do the abstract and introduction summarize the paper's main claims?
Left blank.

☐ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☐ Did you use or create scientific artifacts?

Left blank.

☐ B1. Did you cite the creators of artifacts you used?
Left blank.

☐ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Left blank.

☐ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
Left blank.

☐ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
Left blank.

☐ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
Left blank.

☐ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
Left blank.

C ☐ Did you run computational experiments?

Left blank.

☐ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Left blank.

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☐ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
Left blank.

☐ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
Left blank.

☐ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Left blank.

D ☐ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

☐ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
Left blank.

☐ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
Left blank.

☐ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
Left blank.

☐ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
Left blank.

☐ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
Left blank.