SMoA: Sparse Mixture of Adapters to Mitigate Multiple Dataset Biases

Yanchen Liu[*1,2], **Jing Yan**[*2], **Yan Chen**[*2], **Jing Liu**[†2], **Hua Wu**[2]
[1] Harvard University [2] Baidu Inc.
yanchenliu@fas.harvard.edu
yanjing09, chenyan22, liujing46, wu_hua@baidu.com

Abstract

Recent studies have shown that various biases exist in different NLP tasks, and over-reliance on these biases can result in poor generalization and low adversarial robustness in models. To address this issue, previous research has proposed several debiasing techniques that effectively mitigate specific biases, but are limited in their ability to address other biases. In this paper, we introduce a novel debiasing method, Sparse Mixture-of-Adapters (SMoA), which can effectively and efficiently mitigate multiple dataset biases. Our experiments on Natural Language Inference and Paraphrase Identification tasks demonstrate that SMoA outperforms both full-finetuning and adapter tuning baselines, as well as prior strong debiasing methods. Further analysis reveals that SMoA is interpretable, with each sub-adapter capable of capturing specific patterns from the training data and specializing in handling specific biases.

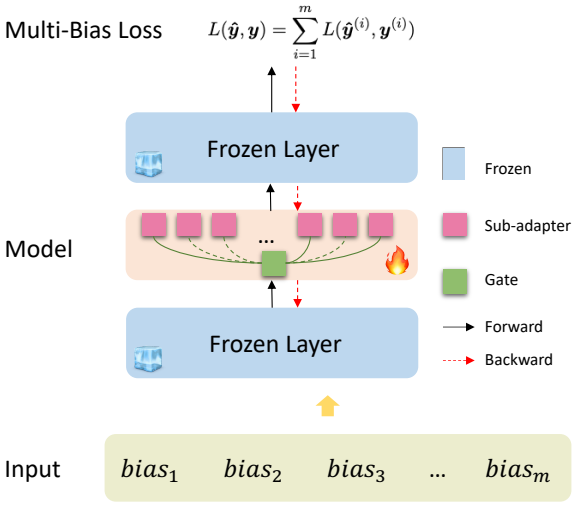

Figure 1: Illustration of our debiasing framework using a Sparse Mixture-of-Adapters (SMoA) architecture, capable of effectively mitigating multiple biases simultaneously. Similar as multi-task learning, we define the loss function of *multi-bias mitigating* as $L(\hat{\boldsymbol{y}}, \boldsymbol{y}) = \sum_{i=1}^{m} L(\hat{\boldsymbol{y}}^{(i)}, \boldsymbol{y}^{(i)})$.

1 Introduction

Recent studies have shown that various biases exist in existing datasets across different tasks, such as Natural Language Inference (NLI)(Williams et al., 2018a; Gururangan et al., 2018; McCoy et al., 2019; Nie et al., 2019; Liu et al., 2020a), Paraphrase Identification (ParaI)(Zhang et al., 2019; Nighojkar and Licato, 2021; Du et al., 2022), and Machine Reading Comprehension (MRC)(Sugawara et al., 2020). Models trained on these biased datasets tend to rely on shallow patterns instead of comprehending the tasks, resulting in poor generalization ability and low adversarial robustness(Geirhos et al., 2020). However, most existing debiasing approaches can only address a specific bias and are not effective in mitigating multiple biases simultaneously (Liu et al., 2020b). These approaches may improve performance on a specific adversarial set, but not on other adversarial sets, and can even harm the model's generalization ability (as discussed in Sec.3). While some works have proposed ensembling different debiased models to enhance the overall generalization ability (Wu et al., 2020; Liu et al., 2020b), this method requires significant computational resources, making it inefficient for real-world applications. Therefore, it is essential to develop a debiasing technique that can handle various biases of different dimensions without incurring excessive costs.

In this paper, we propose a debiasing framework to effectively and efficiently mitigate multiple dataset biases. Specifically, our framework is designed to: 1) *effectively* handle multiple biases simultaneously, improving performance on all adversarial sets as good as or better than only tuning with its own adversarial training data, and

*Equal contribution. Work done during Yanchen Liu's internship at Baidu. †Correspondence to: Jing Liu <liujing46@baidu.com>

2) *efficiently* reduce computational cost compared to model ensemble methods. To achieve these goals, we propose the Sparse Mixture-of-Adapters (SMoA) based on adapter modules proposed by Houlsby et al. (2019). SMoA retains the original parameters of the backbone model fixed and supports multiple debiasing-parameters infused into the backbone, as illustrated in Fig 1.

To effectively mitigate multiple biases, we treat the scenario of dealing with multiple biases of different dimensions in one task simultaneously as similar to multi-task learning, which we refer to as multi-bias mitigating. Furthermore, we insert multiple sub-adapters into the backbone model to learn debiasing knowledge against different biases. These sub-adapters act as specialized experts to handle specific biases. SMoA outputs different representations for different types of debiasing knowledge without tuning the backbone model, which helps retain previously learned information while learning new information and preserves the model's generalization ability. To ensure efficiency in real-world applications, we employ *sparse-gate* to select sub-adapters, which consumes the same computational cost as the *Adapter* method in the inference phase. Compared to tuning the entire backbone model, SMoA only tunes around ∼3.57% of total parameters (i.e., only the selected sub-adapters and the sparse gates are tuned with the mixture of adversarial sets for different biases).

Our experiments on two of the most studied NLP tasks, NLI and ParaI, show that SMoA can effectively improve model robustness in multiple bias dimensions. Our method outperforms previous debiasing approaches while only tuning approximately 3.57% of total parameters, as discussed in Sec. 3. In summary, our contributions are as follows:

1) We propose a novel framework SMoA, which can handle multiple biases simultaneously by learning from multiple adversarial datasets, as described in Sec.2.

2) Our experiments demonstrate the effectiveness and efficiency of SMoA. For the NLI task, SMoA outperforms two-stage full fine-tuning by 1.22% on average, achieving up to 1.08% improvement for `hypothesis-only bias` (Liu et al., 2020a), 0.61% for `inter-sentences bias` (McCoy et al., 2019), and 3.04% for `lexical feature bias` (Nie et al., 2019), while tuning only about 3.57% of parameters. Similar results are shown on the ParaI task. Our analysis of model prediction behaviors indicates that our trained sub-adapters can specialize to handle specific biases, as discussed in Sec.3.

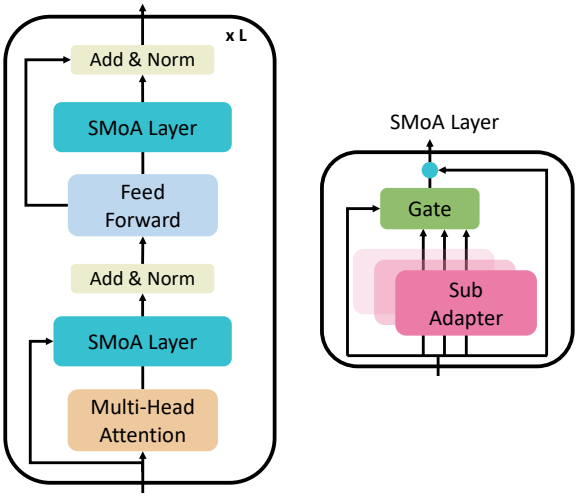

Figure 2: An illustration of the SMoA model's architecture, which consists of a sparse gating network (Shazeer et al., 2017) that controls a mixture-of-adapter layer with n sub-adapters after each multi-head attention and feed-forward network layer.

2 Sparse Mixture of Adapters

In this section, we first introduce three common types of biases in Natural Language Understanding (NLU) tasks. We then present the Sparse Mixture-of-Adapters (SMoA), which is designed to effectively and efficiently mitigate multiple biases.

2.1 Biases in NLU Tasks

Intuitively, a model trained on a basic dataset has the basic competency to solve the fundamental NLP task but may lack some fine-grained sub-competencies, leading to poor performance on out-of-domain or adversarial datasets. This deficiency in sub-competencies often manifests as a bias and can be seen as a "Gordian Knot" in an NLP task. We investigate the typical biases in NLU tasks and classify them into the following three types:

1) **Lexical feature bias**: Previous works (Nie et al., 2019; Gardner et al., 2021; Du et al., 2022) have shown that spurious correlations between words and labels exist in many datasets. Models fine-tuned on these datasets

tend to over-rely on the lexical features of examples rather than understanding the tasks, and assign labels highly correlated with specific biased words in these examples without comprehending the sentence meaning.

2) **Partial-input only bias**: Previous works have shown that a large number of examples in a dataset can be solved using only partial input. For example, Gururangan et al. (2018); Liu et al. (2020a) found that models can perform surprisingly well with only the hypothesis accessible on SNLI (Bowman et al., 2015) and MNLI (Williams et al., 2018b), referring to this phenomenon as *hypothesis-only bias*. In addition, some works have shown that models for Machine Reading Comprehension (MRC) also exploit similar heuristics. (Sugawara et al., 2020) showed that most questions in MRC datasets that are already correctly answered by the model do not necessarily require grammatical and complex reasoning, but only rely on content words or partial context.

3) **Inter-sentences bias**: Model's over-reliance on inter-sentences overlap heuristics has been widely studied in sentence (text) pair tasks. For instance, McCoy et al. (2019) demonstrated that models trained on MNLI (Williams et al., 2018b) may adopt three fallible syntactic heuristics: lexical overlap heuristic, sub-sequence heuristic, and the constituent heuristic. Similar heuristics have also been found by (Zhang et al., 2019) in Paraphrase Identification tasks. They showed that models often predict "Paraphrase" for a sentence pair based on high lexical overlap.

2.2 Sparse Mixture-of-Adapters

Model architecture. Recent works have demonstrated that *Parameter-Efficient Tuning* (PET) brings significant improvements in multi-task learning scenarios (Pfeiffer et al., 2021a). Dealing with different biases simultaneously can be seen as a similar scenario to multi-task learning, and we refer to it as *multi-bias mitigating*. To mitigate multiple biases simultaneously, we propose the Sparse Mixture-of-Adapters (SMoA) method. SMoA incorporates a mixture-of-adapter layer (consisting of n sub-adapters) after each multi-head attention and feed-forward network layer, which is controlled by a sparse gating network (Shazeer et al., 2017).

The model architecture is illustrated in Fig. 2. In a transformer-based backbone model (Vaswani et al., 2017), a SMoA layer $\mathcal{A}$ is inserted after each multi-head attention and feed-forward network layer. SMoA layer $\mathcal{A}$ consists of n sub-adapters ($\mathcal{A} = \{a_1, a_2, ..., a_n\}$), and the sub-adapters are controlled by a sparse gating network. Compared to non-sparse gating (Jordan and Jacobs, 1993), a sparse gating network multiplies the input x by a trainable weight matrix W_g, keeps only the top k values to ensure sparsity, and then applies the softmax function $\sigma(\cdot)$, which can be denoted as follows:

$$G(x) = \sigma(topK(xW_g, k)) \tag{1}$$

where

$$topK(v, k)_i = \begin{cases} v_i & \text{if } v_i \text{ is top } k \text{ element of } v, \\ -\infty & otherwise. \end{cases} \tag{2}$$

For a SMoA layer, given an input x from the previous layer, the sparse gating network computes the weight for each sub-adapter and only keeps the top k values. The top k sub-adapters with their corresponding weights are then used to compute the output of the layer, which is a weighted sum of the outputs of the selected sub-adapters:

$$O_{SMoA} = \sum_{i=1}^{n} G(x)_i a_i(x) \tag{3}$$

where n is the number of sub-adapters. Then the output will be passed into next layer with Add&Norm (He et al., 2016; Ba et al., 2016). [1]

Multi-bias loss. Similar as multi-task learning, the loss function of *multi-bias mitigating* is defined as

$$L(\hat{\boldsymbol{y}}, \boldsymbol{y}) = \sum_{i=1}^{m} L(\hat{\boldsymbol{y}}^{(i)}, \boldsymbol{y}^{(i)}) \tag{4}$$

where m adversarial datasets against m biases are learned. $\boldsymbol{y}^{(i)}$ and $\hat{\boldsymbol{y}}^{(i)}$ represent labels and predictions for m-th datasets respectively.

Debiasing Training. During the training phase, we use the model finetuned on basic datasets as the backbone model M. We freeze the parameters of M and only fine-tune the inserted SMoA layers with a mixture of adversarial datasets against different biases. By freezing the parameters of the

[1] In practice, if there are m adversarial datasets against m biases respectively, we set the number of sub-adapters as $2m - 1$ and k as 2 (if m is more than 2).

backbone model, we can avoid substantial changes to the model's parameters and retain knowledge from previous basic datasets as much as possible. Moreover, training different parts of the model for different biases allows us to effectively deal with multiple biases at the same time.

3 Experiments

To examine the effectiveness and efficiency of SMoA, we apply this framework two of the most studied NLU tasks, NLI and ParaI. We first focus on the NLI task, where several works on bias analysis and debiasing methods have been proposed. In Sec.3.3, we experiment with the ParaI task, which previous works have shown suffers from inter-sentence bias. In Sec.4.1, we discuss the parameter efficiency of SMoA. In Sec.4.2, we conduct a behavior analysis to examine how sub-adapters work to handle different biases. Finally, in Sec.4.3, we compare SMoA with other debiasing methods.

3.1 Experimental Setup

Baselines. We consider two baseline approaches for mitigating biases with adversarial data. The first approach is a two-stage learning approach, which involves further fine-tuning a model that has already been trained on the basic training set with adversarial data for a specific bias. This approach can also be extended sequentially with multiple adversarial datasets to address multiple biases. We refer to this approach as TWO STAGES LEARNING. The second baseline approach is a one-stage learning approach, which involves direct fine-tuning of the model using a training dataset that consists of both basic and adversarial data. We refer to this approach as ONE STAGE LEARNING. We conduct experiments to compare the effectiveness of these two approaches with our proposed SMoA method.

Training Details. We employ *RoBERTa-base* model as the backbone for both NLI and ParaI tasks. We consider three different biases for each task and set the number of sub-adapters n to five, retaining the top 2 values for each SMoA layer. The model is trained using a learning rate of 0.0003, the AdamW optimizer, and a linear learning rate scheduler for five epochs. The evaluation of the model's performance is based on the accuracy metric.

3.2 Natural Language Inference

The Natural Language Inference (NLI) task involves assessing whether a hypothesis h is entailed by a given premise p (Bowman et al., 2015). Previous studies have identified various biases that exist within NLI datasets, including `hypothesis-only bias` (Gururangan et al., 2018; Williams et al., 2018a; Liu et al., 2020a), `inter-sentence bias` (McCoy et al., 2019), and `lexical feature bias` (Nie et al., 2019; Du et al., 2022). In this subsection, we conduct experiments to evaluate the ability of SMoA to effectively mitigate these biases within NLI simultaneously.

3.2.1 Data

Training data. For our experiments on NLI, we augment the MNLI (Williams et al., 2018b) basic dataset with three adversarial datasets that target different biases: `<HARD>`, designed to mitigate the `hypothesis-only bias` and filtered from the MNLI training set; `<LLS>`, which targets `lexical feature bias` and is also filtered from the MNLI training set; and `<HANS>` (McCoy et al., 2019), which targets `inter-sentence bias`.

Evaluation data. We evaluate our models on the NLI adversarial benchmark (Liu et al., 2020b) to obtain an overview of their abilities. The benchmark includes four different test sets, each designed to evaluate a specific type of bias: PI-CD and PI-SP for `hypothesis-only bias`, IS-SD for `inter-sentences bias`, IS-CS for `lexical feature bias`, and ST for both `inter-sentences` and `lexical feature bias`. We provide details on the construction, statistics, and other information of the benchmark in Appendix A.1.

3.2.2 Experimental Results

Our experiments were conducted using the *RoBERTa-base* model (Liu et al., 2019). As described in Section 3.1, we compared two ways of model learning on basic training sets and adversarial sets: TWO STAGES LEARNING and ONE STAGE LEARNING. The experimental results on the NLI adversarial test benchmark and MNLI in-domain test set are presented in Table 1.

Two Stages Learning. In the TWO STAGES LEARNING approach, we first train the pre-trained *RoBERTa-base* model on the basic training dataset

	PI-CD	PI-SP	IS-SD	IS-CS	LI-LI	LI-TS	ST	Avg.	MNLI (id test)
Roberta-base Baseline	75.87	79.51	71.53	71.8	89.33	85.04	72.43	77.93	87.44
TWO STAGES LEARNING									
full tuning	75.22	83.29	86.33	74.70	86.11	85.53	70.37	80.22	85.06
adapter tuning	76.17	83.02	85.11	74.54	**90.55**	85.70	71.05	80.88	86.50
SMOA	75.44	84.37	86.94	**77.74**	89.77	85.70	70.11	**81.44**	85.15
ONE STAGE LEARNING									
full tuning	76.39	81.40	85.86	73.17	88.20	85.68	71.79	80.36	87.47
adapter tuning	75.41	81.13	86.26	72.87	90.40	85.24	**72.75**	80.58	87.42
SMOA	**76.44**	81.67	86.23	74.54	89.17	85.36	72.45	80.84	**87.87**
Ablation Experiment									
(MNLI) + <HARD>	75.04	**85.98**	74.67	74.39	90.09	**85.85**	67.37	79.06	83.91
(MNLI) + <HANS>	72.71	78.98	**87.26**	70.12	88.78	84.59	71.63	79.15	85.67
(MNLI) + <LLS>	74.98	77.09	75.49	72.10	87.62	84.60	69.94	77.40	86.56
Model Ensemble	74.89	81.40	77.56	73.02	89.44	**85.85**	70.67	78.98	86.87

Table 1: Results of RoBERTa-base model on the NLI adversarial test benchmark proposed by (Liu et al., 2020b) and in-domain MNLI test set. **Bold face** and underlined indicate the first and second highest performance on each test set.

MNLI, and the finetuned model is used as the backbone. Next, we combine the three augmented datasets (<HARD> , <LLS> , and <HANS>) and train the model using one of three approaches: **full tuning** (tuning the entire RoBERTa-base model), **adapter tuning** (only tuning the inserted adapter layers), and **SMOA** (only tuning the inserted SMOA layers) with the mixed adversarial datasets.

As shown in Table 1 under TWO STAGES LEARNING, the proposed SMOA method outperforms the baselines (i.e., only finetuning *RoBERTa-base* with the MNLI training set), full tuning, and adapter tuning methods across almost all categories of the NLI adversarial test benchmark. Specifically, SMOA significantly outperforms the baseline across all categories except for a slight decrease on PI-CD and ST, yielding an average gain of 3.5%. Compared to full tuning and adapter tuning, SMOA produces an average improvement of 1.22% and 0.56% respectively. Notably, compared to full tuning, SMOA shows an improvement of 1.08% on PI-SP (`hypothesis-only bias`), 0.61% on IS-SD (`inter-sentences bias`), and 3.04% on IS-CS (`lexical features bias`). Overall, SMOA yields an average gain of 1.22% (column Avg.) while only tuning 3.57% of the parameters (as discussed in Sec. 4.1), demonstrating that the proposed framework is capable of mitigating multiple biases simultaneously in the NLI task.

One Stage Learning. We compared the experimental results of TWO STAGES LEARNING with ONE STAGE LEARNING, which involves training with the original dataset and multiple adversarial datasets all at once. As shown in Tab. 1, for adapter tuning and SMOA on the NLI adversarial benchmark, the TWO STAGES LEARNING strategy outperforms ONE STAGE LEARNING; for full tuning, ONE STAGE LEARNING is slightly better. Furthermore, when using the ONE STAGE LEARNING strategy, SMOA brings considerable improvements (0.48% and 0.25%) compared to full tuning and adapter tuning. Regarding the in-domain performance, as shown in column MNLI, although TWO STAGES LEARNING brings larger improvement on the NLI adversarial test benchmark, SMOA with ONE STAGE LEARNING demonstrates better performance on the in-domain MNLI test set.

Ablation experiment. To analyze the effect of each adversarial training set individually, we trained the entire MNLI-trained model with the <HANS> , <HARD> , and <LLS> datasets separately. The ablation experiments presented in Table 1 demonstrate that each augmented dataset brings significant improvements on their corresponding bias: training with <HARD> brings a 6.47% improvement on the PI-SP subsection, <HANS> brings a 15.73% improvement on the IS-SD subsection, and <LLS> brings a 0.3% improvement on the IS-CS subsection. It is worth noting that training with <HARD> demonstrates greater improvement on the IS-CS subsection than <LLS> , the augmented dataset targeting only `lexical feature bias`. However, an improvement in a specific bias can result in a decrease in other biases.

As shown in the Avg. column of Table 1, single-tuning performs poorly on average performance, and SMoA performs better overall. Moreover, we ensemble the above three full-tuned models and used majority voting for predictions. The ensemble model shows better performance on the in-domain test set compared to single-tuning, but its performance on adversarial test sets is not as good.

In general, SMoA can address multiple biases simultaneously and shows significant improvements on adversarial robustness evaluation sets while tuning fewer parameters.

3.3 Paraphrase Identification

Furthermore, we consider the Paraphrase Identification (ParaI) task, which involves determining if a given sentence pair ($s1$ and $s2$) carries the same meaning. However, previous studies, such as (Zhang et al., 2019; Nighojkar and Licato, 2021), have highlighted the limitations of existing ParaI datasets, which lack sentence pairs with high lexical and syntactic overlap that are not paraphrases, as well as pairs with low overlap that are paraphrases. As a result, models trained on these datasets tend to predict sentence pairs with high overlap as "Paraphrase" and those with low overlap as "Non-paraphrase," leading to failures in real-world scenarios. Moreover, (Du et al., 2022) revealed that ParaI models suffer from `lexical feature bias`, where models rely too heavily on spurious relations between labels and words.

3.3.1 Data

Training Data. For the basic training of our ParaI models, we use the *Quora Question Pairs* (QQP) dataset[2]. For adversarial training, we use three adversarial datasets proposed by (Zhang et al., 2019; Nighojkar and Licato, 2021; Du et al., 2022): PAWS (against `high-overlap bias`), APT (against `low-overlap bias`), and LLS (against `lexical feature bias`).

Evaluation Data. We evaluate our models on four evaluation datasets: PAWS_{QQP}, $\text{PAWS}_{\text{wiki}}$, APT and LLS.

As there is no available evaluation dataset for models' compositionality sensitivity in ParaI task, we follow (Du et al., 2022) and filter out unbiased examples from the QQP training set as the LLS training set, and unbiased examples from the QQP test set as the LLS evaluation set. For more detailed data statistics on the adversarial ParaI datasets, please refer to Appendix A.2.

[2]https://data.quora.com/First-Quora-Dataset-Release-Question-Pairs

3.3.2 Experimental Results

We use a pre-trained *RoBERTa-base* model that has been fine-tuned on the QQP dataset (Liu et al., 2019) as the backbone model for our SMoA method, and we train it with the augmented dataset consisting of the three adversarial datasets mentioned above (Two Stages Learning). Similar as Sec. 3.2.2, we compare our method with full tuning and adapter tuning. Tab. 2 presents the experimental results of *RoBERTa-base* on ParaI adversarial test benchmark and in-domain QQP test set. Our results indicate that SMoA outperforms baseline on all adversarial evaluation sets, with accuracy improvements ranging from 1.77% to 53.03%. In comparison with full tuning and adapter tuning, SMoA yields better performance on PAWS_{QQP} and APT by 0.44% and 1.11%, respectively. However, all three strategies exhibit poor performance on in-domain QQP test set and LLS evaluation set, probably because the original QQP dataset does not suffer from lexical feature bias to a significant extent, resulting in similar distributions between the LLS evaluation set and the QQP test set.

4 Analysis

4.1 Parameter Efficiency Analysis

The experimental results have demonstrated that SMoA is an effective method for addressing multiple biases simultaneously. Here we examine the parameter efficiency of SMoA. As shown in Tab. 6, SMoA only tunes 3.70% of the parameters compared to full tuning. Furthermore, when compared to debiasing methods that rely on model ensembling, SMoA exhibits even greater gains in parameter efficiency.

4.2 Sub-adapters' Behavior Analysis

We hypothesize that each trained sub-adapter is capable of capturing specific patterns from the training data and learning to handle specific biases. To verify this hypothesis, we analyze the distribution of the top 2 selected sub-adapters for each adversarial subset of the NLI adversarial benchmark (Liu et al., 2020b). We calculate the distribution for each SMoA layer of our trained model in Section 3.2 and observe that the distribution for each adver-

	PAWS_{QQP}	$\text{PAWS}_{\text{WIKI}}$	APT	LLS	Avg.	QQP (id test)
RoBERTa-base Baseline	39.14	47.11	69.94	**91.49**	61.92	**91.47**
TWO STAGES LEARNING						
full tuning	89.66	81.36	77.16	88.41	84.14	88.19
adapter tuning	91.73	81.70	77.16	88.98	84.89	87.28
SMOA	**92.17**	**81.71**	**78.27**	88.05	**85.05**	87.72

Table 2: For ParaI task, we train on QQP finetuned model with augmented datasets composed of PAWS_{QQP}, APT and LLS train sets using three different strategies: full tuning and adapter tuning and SMOA. The table shows the accuracy on in-domain test set QQP, and PAWS_{QQP}, APT, LLS, $\text{PAWS}_{\text{WIKI}}$ evaluation sets.

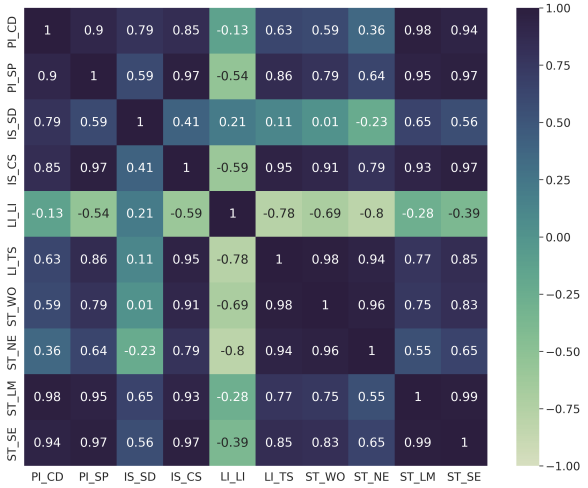

Figure 3: The Pearson's correlation coefficients of top 2 sub-adapters distribution of the last SMOA layer between different adversarial subsets of the NLI adversarial benchmark (Liu et al., 2020b).

sarial subset is nearly identical in the low layers. However, in the high layers, the distribution for each adversarial subset differs from one another and exhibits correlations.

We present the Pearson's correlation coefficients of the sub-adapter distribution in the last layer for each adversarial subset in Figure 3. Our analysis reveals that the distributions of selected sub-adapters exhibit higher correlation scores for subsets that have higher similarities. For instance, the correlation coefficient is up to 0.9 for subsets PI-CD and PI-SP, both of which are constructed with Partial-input (PI) heuristics. In contrast, the correlation score of selected sub-adapters is low, at -0.23, between subsets IS-SD and ST-NE, which have a surrogate correlation score of only 32.9/100, as reported by Liu et al. (2020b).

The experimental results indicate that examples with similar biases tend to be handled by the same set of sub-adapters. This suggests that SMOA not only adds more parameters but can also specialize in handling specific biases.

4.3 Comparison with other Debiasing Methods

We conducted further experiments on the NLI task using a *BERT-base* model as the backbone to compare the effectiveness of SMOA with previous methods. As presented in Table 3, the results demonstrate that when using a *BERT-base* model that has been fine-tuned on MNLI as the backbone and further training it on the adversarial datasets using the TWO STAGES LEARNING, SMOA significantly outperforms the baseline across all categories except for a minor decrease on ST, with an average gain of 6.9%. Moreover, when compared to full tuning and adapter tuning, SMOA yields better performance on PI-CD, PI-SP, IS-SD, and IS-CS, with an average gain of 1.24% and 1.5%, respectively. Notably, SMOA shows a 4.31% improvement on PI-CD (`hypothesis-only bias`), 2.57% on IS-SD (`inter-sentences bias`), and 2.75% on IS-CS (`lexical feature bias`). Surprisingly, SMOA demonstrates negligible drops in in-domain performance for the *BERT-base* model, indicating its ability to mitigate multiple biases simultaneously while preserving the model's in-domain capability.

In Tab. 3, we compare our method with various data-augmentation methods proposed by Liu et al. (2020b). The results demonstrate that SMOA outperforms these methods by an average of 5.28%. These data-augmentation methods are designed to address only a single bias and are comparatively simple. Additionally, (Wu et al., 2022) proposed a data-generation-debiasing method to mitigate spurious correlations. Compared to this method, SMOA shows a better performance of 2.71% on average, particularly 1.74% and 1.98% improvements for `hypothesis-only bias`.

	PI-CD	PI-SP	IS-SD	IS-CS	LI-LI	LI-TS	ST	Avg.	MNLI
BERT-base Baseline	$70.3_{\pm 0.5}$	$73.7_{\pm 1.4}$	$53.5_{\pm 2.3}$	$64.8_{\pm 1.4}$	$85.5_{\pm 0.9}$	$81.6_{\pm 1.4}$	$69.2_{\pm 0.8}$	$71.2_{\pm 0.8}$	83.5
TWO STAGES LEARNING									
full tuning	73.08	75.47	85.84	66.46	86.21	84.04	67.01	76.87	82.52
adapter tuning	73.35	74.12	83.75	67.07	87.31	83.38	67.15	76.59	83.07
SMOA	73.44	79.78	88.41	69.82	85.03	83.39	66.91	78.11	83.41
Data-augmentation heuristics proposed by (Liu et al., 2020b)									
Text Swap	71.7	72.8	63.5	67.4	86.3	86.8	66.5	73.6	83.7
Sub (synonym)	69.8	72.0	62.4	65.8	85.2	82.8	64.3	71.8	83.5
Sub (MLM)	71.0	72.8	64.4	65.9	85.6	83.3	64.9	72.6	83.6
Paraphrase	72.1	74.6	66.5	66.4	85.7	83.1	64.8	73.3	83.7
Prior debiasing strategies									
(Wu et al., 2022)	$71.7_{\pm 0.9}$	$77.8_{\pm 1.2}$	$66.9_{\pm 3.7}$	$71.1_{\pm 0.7}$	$89.1_{\pm 1.0}$	$82.3_{\pm 0.9}$	$69.3_{\pm 0.8}$	$75.4_{\pm 0.8}$	82.70

Table 3: Results on the adversarial benchmark proposed by (Liu et al., 2020b). We compared our method with full tuning and adapter tuning, data augmentation as well as other prior debiasing strategies. Due to experiments' compatibility, we directly report the results from previous works.

5 Related Work

Bias Analysis and Evaluation. Spurious correlations in existing datasets have been extensively studied across various NLP tasks. For example, previous work has demonstrated that NLI systems suffer from hypothesis-only bias (Gururangan et al., 2018; Liu et al., 2020a), inter-sentence bias (McCoy et al., 2019), and lexical features bias (Nie et al., 2019). For paraphrase identification, biases have also been widely studied (Zhang et al., 2019; Nighojkar and Licato, 2021). In addition, some work has attempted to analyze spurious correlations from a theoretical perspective (Gardner et al., 2021; Du et al., 2022).

Debiasing Methods. Previous debiasing methods have addressed biases from either the model level or the dataset level. At the model level, Teney et al. (2020) introduced a new training objective that utilizes counterfactual examples for debiasing, while Belinkov et al. (2019a,b); Zhou and Bansal (2020) propose to learn less biased representations for input. Clark et al. (2019); He et al. (2019); Karimi Mahabadi et al. (2020) deal with specific biases by ensembling a bias-only model with the main model or weakening the impact of biases via re-weighted training examples given by a bias model. At the dataset level, Sakaguchi et al. (2019); Bras et al. (2020) propose filtering the existing dataset to obtain a debiased one. Furthermore, a series of data augmentation strategies proposed by (Ross et al., 2021; Wu et al., 2021; Wang et al., 2021; Reid and Zhong, 2021; Ross et al., 2020; Liu et al., 2020b) have demonstrated improvements in robustness. Wu et al. (2022) combine data augmentation with dataset filtering to mitigate spurious correlations in existing datasets.

Parameter Efficient Learning. Previous work has shown that Parameter Efficient Learning (He et al., 2021; Ding et al., 2022), including techniques such as prompt tuning (Lester et al., 2021), prefix tuning (Li and Liang, 2021), and adapter tuning (Houlsby et al., 2019), can achieve comparable or better performance and robustness than full fine-tuning, while using significantly fewer parameters (Liu et al., 2021; He et al., 2021; Liu et al., 2022). Pfeiffer et al. (2021a) proposed an adapter-based framework for incorporating knowledge from multiple tasks, achieving significant success in multi-task learning. Strategies for ensembling multiple adapters have been explored in Wang et al. (2022), Asai et al. (2022), Wang et al. (2020), and Pfeiffer et al. (2021b). While previous work has significantly improved model performance on specific sets, the question of whether Parameter Efficient Learning can be used to simultaneously mitigate multiple biases for one task remains largely unexplored.

6 Conclusion

Most existing debiasing methods tend to focus on addressing one specific type of bias, which can result in significant improvements on particular adversarial test sets, but fail on others. However, in real-world applications, multiple biases may exist simultaneously, which requires an effective and efficient approach to mitigate them. To address this issue, we propose the Sparse Mixture-of-Adapters (SMOA) method. Our experimental results on the

Natural Language Inference (NLI) and Paraphrase Identification (ParaI) tasks demonstrate that SMoA outperforms previous debiasing methods across various types of biases. Additionally, we conduct an analysis to examine the interpretability of SMoA and show that the sub-adapters in SMoA can specialize to handle specific biases, providing a more fine-grained understanding of how SMoA mitigates multiple biases.

Limitation

It is important to note that while SMoA has shown promising results in mitigating multiple biases, the addition of sparse gates and the selection of sub-adapters can potentially increase the model's inference time. This trade-off between performance and efficiency must be carefully considered when implementing the SMoA architecture in real-world applications. Additionally, when new bias types are introduced, the SMoA architecture may need to be retrained to effectively handle these new biases, which is a potential limitation of the current approach. In future work, we will explore ways to further improve the effectiveness and efficiency of our debiasing approach. One possible direction for future research is to investigate methods for efficiently updating the SMoA architecture with new bias types, without having to retrain the entire model.

References

Akari Asai, Mohammadreza Salehi, Matthew E. Peters, and Hannaneh Hajishirzi. 2022. Attempt: Parameter-efficient multi-task tuning via attentional mixtures of soft prompts.

Jimmy Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. 2016. Layer normalization. *ArXiv*, abs/1607.06450.

Yonatan Belinkov, Adam Poliak, Stuart Shieber, Benjamin Van Durme, and Alexander Rush. 2019a. 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.

Yonatan Belinkov, Adam Poliak, Stuart Shieber, Benjamin Van Durme, and Alexander Rush. 2019b. On adversarial removal of hypothesis-only bias in natural language inference. In *Proceedings of the Eighth Joint Conference on Lexical and Computational Semantics (*SEM 2019)*, pages 256–262, Minneapolis, Minnesota. Association for Computational Linguistics.

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.

Ronan Le Bras, Swabha Swayamdipta, Chandra Bhagavatula, Rowan Zellers, Matthew E. Peters, Ashish Sabharwal, and Yejin Choi. 2020. Adversarial filters of dataset biases. *CoRR*, abs/2002.04108.

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.

Ning Ding, Yujia Qin, Guang Yang, Fuchao Wei, Zonghan Yang, Yusheng Su, Shengding Hu, Yulin Chen, Chi-Min Chan, Weize Chen, Jing Yi, Weilin Zhao, Xiaozhi Wang, Zhiyuan Liu, Hai-Tao Zheng, Jianfei Chen, Yang Liu, Jie Tang, Juanzi Li, and Maosong Sun. 2022. Delta tuning: A comprehensive study of parameter efficient methods for pre-trained language models.

Yanrui Du, Jing-Fu Yan, Yan Chen, Jing Liu, Sendong Zhao, Huaqin Wu, Haifeng Wang, and Bing Qin. 2022. Less learn shortcut: Analyzing and mitigating learning of spurious feature-label correlation. *ArXiv*, abs/2205.12593.

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 Zemel, Wieland Brendel, Matthias Bethge, and Felix A Wichmann. 2020. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 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.

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.

Junxian He, Chunting Zhou, Xuezhe Ma, Taylor Berg-Kirkpatrick, and Graham Neubig. 2021. Towards a unified view of parameter-efficient transfer learning. *CoRR*, abs/2110.04366.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for NLP. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 2790–2799. PMLR.

M.I. Jordan and R.A. Jacobs. 1993. Hierarchical mixtures of experts and the em algorithm. In *Proceedings of 1993 International Conference on Neural Networks (IJCNN-93-Nagoya, Japan)*, volume 2, pages 1339–1344 vol.2.

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.

Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3045–3059, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation.

Tianyu Liu, Zheng Xin, Baobao Chang, and Zhifang Sui. 2020a. HypoNLI: Exploring the artificial patterns of hypothesis-only bias in natural language inference. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 6852–6860, Marseille, France. European Language Resources Association.

Tianyu Liu, Zheng Xin, Xiaoan Ding, Baobao Chang, and Zhifang Sui. 2020b. An empirical study on model-agnostic debiasing strategies for robust natural language inference. In *Proceedings of the 24th Conference on Computational Natural Language Learning*, pages 596–608, Online. Association for Computational Linguistics.

Xiao Liu, Kaixuan Ji, Yicheng Fu, Weng Lam Tam, Zhengxiao Du, Zhilin Yang, and Jie Tang. 2021. P-tuning v2: Prompt tuning can be comparable to fine-tuning universally across scales and tasks.

Yanchen Liu, Timo Schick, and Hinrich Schütze. 2022. Semantic-oriented unlabeled priming for large-scale language models.

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.

Pasquale Minervini and Sebastian Riedel. 2018. Adversarially regularising neural NLI models to integrate logical background knowledge. In *Proceedings of the 22nd Conference on Computational Natural Language Learning*, pages 65–74, Brussels, Belgium. 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.

Yixin Nie, Yicheng Wang, and Mohit Bansal. 2019. Analyzing compositionality-sensitivity of nli models. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):6867–6874.

Animesh Nighojkar and John Licato. 2021. Improving paraphrase detection with the adversarial paraphrasing task. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 7106–7116, Online. Association for Computational Linguistics.

Jonas Pfeiffer, Aishwarya Kamath, Andreas Rücklé, Kyunghyun Cho, and Iryna Gurevych. 2021a. AdapterFusion: Non-destructive task composition for transfer learning. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 487–503, Online. Association for Computational Linguistics.

Jonas Pfeiffer, Aishwarya Kamath, Andreas Rücklé, Kyunghyun Cho, and Iryna Gurevych. 2021b. Adapterfusion: Non-destructive task composition for transfer learning. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 487–503.

Machel Reid and Victor Zhong. 2021. LEWIS: levenshtein editing for unsupervised text style transfer. *CoRR*, abs/2105.08206.

Alexis Ross, Ana Marasovic, and Matthew E. Peters. 2020. Explaining NLP models via minimal contrastive editing (mice). *CoRR*, abs/2012.13985.

Alexis Ross, Tongshuang Wu, Hao Peng, Matthew E. Peters, and Matt Gardner. 2021. Tailor: Generating and perturbing text with semantic controls. *CoRR*, abs/2107.07150.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2019. WINOGRANDE: an adversarial winograd schema challenge at scale. *CoRR*, abs/1907.10641.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. 2017. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer.

Saku Sugawara, Pontus Stenetorp, Kentaro Inui, and Akiko Aizawa. 2020. Assessing the benchmarking capacity of machine reading comprehension datasets. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):8918–8927.

Damien Teney, Ehsan Abbasnejad, and Anton van den Hengel. 2020. Learning what makes a difference from counterfactual examples and gradient supervision. *CoRR*, abs/2004.09034.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Dong Wang, Ning Ding, Piji Li, and Haitao Zheng. 2021. CLINE: Contrastive learning with semantic negative examples for natural language understanding. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 2332–2342, Online. Association for Computational Linguistics.

Haohan Wang, Da Sun, and Eric P. Xing. 2018. What if we simply swap the two text fragments? A straightforward yet effective way to test the robustness of methods to confounding signals in nature language inference tasks. *CoRR*, abs/1809.02719.

Ruize Wang, Duyu Tang, Nan Duan, Zhongyu Wei, Xuanjing Huang, Jianshu Ji, Guihong Cao, Daxin Jiang, and Ming Zhou. 2020. K-adapter: Infusing knowledge into pre-trained models with adapters. In *Findings*.

Yaqing Wang, Subhabrata Mukherjee, Xiaodong Liu, Jing Gao, Ahmed Hassan Awadallah, and Jianfeng Gao. 2022. Adamix: Mixture-of-adapter for parameter-efficient tuning of large language models. *arXiv preprint arXiv:2205.12410*.

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

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

Mingzhu Wu, Nafise Sadat Moosavi, Andreas Rücklé, and Iryna Gurevych. 2020. Improving qa generalization by concurrent modeling of multiple biases. *arXiv preprint arXiv:2010.03338*.

Tongshuang Wu, Marco Túlio Ribeiro, Jeffrey Heer, and Daniel S. Weld. 2021. Polyjuice: Automated, general-purpose counterfactual generation. *CoRR*, abs/2101.00288.

Yuxiang Wu, Matt Gardner, Pontus Stenetorp, and Pradeep Dasigi. 2022. Generating data to mitigate spurious correlations in natural language inference datasets. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2660–2676, Dublin, Ireland. Association for Computational Linguistics.

Yuan Zhang, Jason Baldridge, and Luheng He. 2019. PAWS: Paraphrase Adversaries from Word Scrambling. In *Proc. of NAACL*.

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.

A Dataset Details

A.1 Natural Language Inference

1. Adversarial training sets

HARD and *LLS* are filter in MNLI training set, and *HANS* is proposed in (McCoy et al., 2019). Data statistics are provided in Tab. 4.

1) **HARD**: To against **hypothesis-only bias** in NLI, as (Gururangan et al., 2018; Williams et al., 2018a), we use a finetuned external model (*RoBERTa-large* (Liu et al., 2019)) to filter the examples in original MNLI training set that are not correctly classified if only hypothesis is provided to construct a adversarial dataset *HARD*.[3]

2) **LLS**: As (Du et al., 2022), who define the word highly co-occurring with a specific label as *biased word* of a dataset, we analysis the co-occurrence of words in MNLI training set and specific labels to obtain the *biased words*, and filter out the examples not containing any biased words ("unbiased examples") as *LLS*.[4] We expect to mitigate model's over-reliance on the **lexical feature bias** by incorporating the debiased dataset *LLS* into training.

3) **HANS**: Except above two augmented datasets filtered from original training set, we consider an existing dataset *HANS* (McCoy et al., 2019). *HANS* provides a training set and an evaluation set. HANS evaluation set includes premise-hypothesis pairs with high lexical, sub-sequence and constituent overlap but not semantically entailable where model's overlay heuristics fail. We use *HANS* training set, and aim to mitigating model's **inter-sentences bias**. In their original training set, there are only two labels (entailment and non-entailment), because they think the distinction between contradiction and neutral was often unclear for their cases. In order to use this dataset for training a three-class classification model, we label the "non-entailment" subset with an external large finetuned model, filter out as "entailment" incorrectly classified examples, merge with "entailment" subset and balance the label distribution.

2. Adversarial test sets We use NLI adversarial test benchmark (Liu et al., 2020b) as test sets.

[3]In the original MNLI training set, there are many examples cannot be correctly classified with only hypothesis. We randomly select 40500 label balanced examples as train set and 4500 as dev set.

[4]We choose the words with frequency ≥ 3 and most possible label ratio frequency ≥ 0.385 as biased word. We balance the unbiased examples set and split it as train and dev subset.

	HARD	HANS	LLS
train	40500	3261	37065
dev	4500	363	4119

Table 4: Data statistics of adversarial datasets of NLI.

1) **Partial-input (PI) heuristics**:

PI-CD: a subset of SNLI test set built by (Gururangan et al., 2018) aiming to hypothesis-only bias, also known as SNLI-hard.

PI-SP: a subset of MultiNLI mismatched dev set built by (Liu et al., 2020a) aiming to surface patterns heuristics in NLI.

2) **Inter-sentences (IS) heuristics**:

IS-SD: syntactic diagnostic dataset HANS (McCoy et al., 2019).

IS-CS: (Nie et al., 2019) compute the 'lexically misleading scores (LMS)' for each instance in the SNLI test and MNLI dev sets using a softmax regression model to measure the importance of compositional information (not only lexical features) for solve this example. IS-CS is the subset whose LMS are larger that 0.7.

3) **Logical-inference (LI) ability**:

LI-LI: lexical inference test dataset by (Glockner et al., 2018; Naik et al., 2018).

LI-TS: adversarial examples created by swapping the premise and hypothesis aiming to first-order logical inference ability (Minervini and Riedel, 2018; Wang et al., 2018).

4) **Stress test (ST)**: An aggregation of "word overlap", "negation" , "length mismatch" and "spelling errors" tests in (Naik et al., 2018).

A.2 Paraphrase Identification

Tab. 5 presents the data statistics of adversarial datasets of ParaI task, PAWS_{QQP}, APT and LLS. For PAWS_{QQP} and APT, we split the original train sets as our train and dev subsets.

	PAWS_{QQP}	APT	LLS
train	10788	3370	9000
dev	1200	376	1000
test	677	1262	1398

Table 5: Statistics of adversarial datasets of ParaI task.

B Parameter Efficiency Analysis

Tab. 6 demonstrates comparison of the number of parameters and forward time of RoBERTa-base and SMoA.

	SMoA	RoBERTa-base	Ratio
#total parameters	129258939	124647939	1.0370
#trainable parameters	4611000	124647939	0.0370
trainable ratio	3.57%	100%	0.0357
forward time (s)	0.2356	0.0863	2.7300

Table 6: Comparison of the number of parameters and forward time of vanilla RoBERTa-base model and SMoA (with 5 sub-adapters). The inference time is tested using a single A100-SXM4-40GB.