Causal-Guided Active Learning for Debiasing Large Language Models

Anonymous ACL submission

Abstract

Although achieving promising performance, recent analyses show that current generative large language models (LLMs) may still capture dataset biases and utilize them for generation, leading to poor generalizability and harmfulness of LLMs. However, due to the diversity of dataset biases and the over-optimization problem, previous prior-knowledge-based debiasing methods and fine-tuning-based debiasing methods may not be suitable for current LLMs. To address this issue, we explore combining active learning with the causal mechanisms and propose a casual-guided active learning (CAL) framework, which utilizes LLMs itself to automatically and autonomously identify informative biased samples and induce the bias patterns. Then a cost-effective and efficient in-context learning based method is employed to prevent LLMs from utilizing dataset biases during generation. Experimental results show that CAL can effectively recognize typical biased instances and induce various bias patterns for debiasing LLMs.

1 Introduction

Large language models (LLMs) are growing to be the foundation of Natural Language Processing. Through the generative pretraining process upon a large-scale corpus, the LLMs have demonstrated impressive performance in understanding the language and conducting complex reasoning tasks (Achiam et al., 2023), demonstrating immense potential in real-world applications.

However, the generative pretraining process is a double-edged sword, as it would also inevitably incur **dataset bias** into the LLMs such as position bias and stereotype bias (Schick et al., 2021; Navigli et al., 2023; Zheng et al., 2023; Shaikh et al., 2023). This is because, the LLMs only *passively* learn to model the *correlation* between contexts in the pretraining corpus, and the pretraining corpus is biased as it reflects the inherent preference or prejudice of human beings. For example, the existence of position bias is due to the subconscious human belief that the first option is better, leading to a higher frequency of the first option in corpora, and LLMs trained to model the corpus distribution would also capture such biased correlation. Such biases would lead to *poor generalizability* and *harmfulness* of LLMs (Navigli et al., 2023; Huang et al., 2023). For instance, when an LLM is asked to evaluate which option is better, the LLM may utilize position bias and tend to choose the first option. However, which option is better is completely unrelated to its position. Therefore, when the second option is generally better in some datasets, the performance of the LLM will significantly decline. While biases such as stereotyping bias would make LLMs generate harmful content such as women are less capable in STEM fields, which in turn reinforces harmful stereotypes.

These problems highlight the necessity of debiasing LLMs. The key issue to debias LLMs lies in how to recognize the dataset biases and prevent it from utilizing biases during inference. To this end, prevalent methods rely on researchers' prior knowledge to artificially recognize the potential dataset biases, and then eliminate such biases through aligning or prompt-based regularization (Schick et al., 2021; Oba et al., 2023; Liu et al., 2023). However, due to the diversity and complexity of dataset biases (Poliak et al., 2018; Schuster et al., 2019; Schick et al., 2021), it's unpractical to identify them one by one manually. A vast amount of biases remains unrecognized in different tasks (Nie et al., 2020) and new biases are continually being discovered.

Hence, there is an urgent need for methods to automatically identify biases of generative LLMs. However, previous automatic debiasing methods are mainly designed for discriminative models and are hard to adapt to generative LLMs. Moreover, these methods generally rely on a fine-tuning-

based process on certain dataset(s) to regularize the model. The finetuning-based debiasing process would lead to over-optimization and undermine the generalizability of LLMs on other tasks.

To address these issues, considering the powerful pattern recognition and inductive ability of LLMs, we explore combining *active* learning with the *causal* mechanisms and propose a **C**asual-guided **A**ctive **L**earning (CAL) framework, which utilizes LLMs themselves to automatically and autonomously identify biased samples and induce the bias patterns. Active learning aims at selecting the most informative instances, and then querying external information source(s) to label these data points. In the debiasing scenario, CAL identifies the biased instances by finding instances where the LLMs fail to model *causal invariant* semantic relationship among context, then selects the most informative biased instances by finding the instances on which dataset biases have the most influence on the generation of LLMs. The causal invariance can be employed to disentangle the semantic information with dataset biases, as the content of the subsequent text is decided by the semantics of the preceding text (i.e., "*causal*"), and such relationship exists in all corpora (i.e., "*invariant*"); on the contrary, although the subsequent text would be correlative to dataset bias, such correlation changes upon different datasets. Given the biased instances, a set of explainable bias patterns is further induced, and we devise a cost-effective and efficient in-context learning (ICL) based method to regularize LLMs using the explainable bias patterns.

Experimental results show that our approach can automatically induce various bias patterns (some of them may be unreported), and improve the generalizability and safety of LLMs by using the ICL-based debiasing method based on the bias patterns.

2 Preliminary

2.1 Dataset Bias within Textual Corpus under Causal Perspective

Text records and reflects the thoughts of human beings. Inherent biases such as gender and racial biases persist in the human mind, and thus are also reflected in various corpora (Schick et al., 2021; Navigli et al., 2023). Due to potential annotation artifacts, various biases such as position and verbosity biases still broadly exist in task-specific datasets.

Formally, as shown in Figure 1 (a), given a piece of text X, the subsequent text Y within a corpus $\mathcal{D}$ would be affected by two factors: (1) The semantic relationship between X and Y, (2) The existence of dataset bias within $\mathcal{D}$. For example, given $X =$ **The physician hired the secretary because**, due to the existence of gender bias, the following text Y in the corpus would more likely be **he** was overwhelmed with clients, rather than **she**. Such **biased relationship** characterizes the unwanted correlation between the context brought by dataset bias. In the following sections, for clarity, we denote the semantic relationship as $f_S(\cdot)$, and denote the biased relationship as $g_B(\cdot)$. Hence, given X, the conditional distribution of Y given X in corpus $\mathcal{D}$ can be formalized as $P(Y|X) = P(f_S(X), g_B(X)|X)$.

The key difference between the semantic relationship and the biased relationship is that the semantic relationship possesses the *causal invariance*, while the biased relationship does not. Specifically, for all instances upon all datasets, given preceding text X, the subsequent text Y would be determined by the semantic relationship (Pearl et al., 2000; Pearl, 2009), while the biased relationship only describes certain superficial statistical correlation between X and Y. Consider the example where an LLM acts as a judge to assess the responses of two AI assistants, as illustrated in Figure 1 (a): The answer (Y) is determined by the semantic relationship between the prompt X and answer Y. While in the corpus, certain biases such as the position of the responses that show a correlation with the answer can be predictive (Wang et al., 2023). However, Y is not determined by the bias and such a correlation may fail to be predictive in other instances. Hence, as Y is determined by X, their semantic relationship is a "causal" relationship and invariant upon all instances. While the biased relationship is only correlative.

2.2 Biases of Generative LLMs

During the pretraining and task-specific supervised fine-tuning process, the training objective of generative LLMs is consistent, i.e., learn to generate the subsequent text Y given input text X. given X in corpus $\mathcal{D}$, the distribution of Y can be formalized as $P(Y|X) = P(f_S(X), g_B(X)|X)$, the generative LLMs would inevitably be trained to model both $f_S(X)$ and $g_B(X)$. Therefore, given preceding text X_i, LLMs would not only attend to the semantics of X_i but also would attend to the biased patterns such as negation word, gender indicator, position of choices, etc, to generate Y. As a result, during inference, the model generation $\hat{Y}$ would

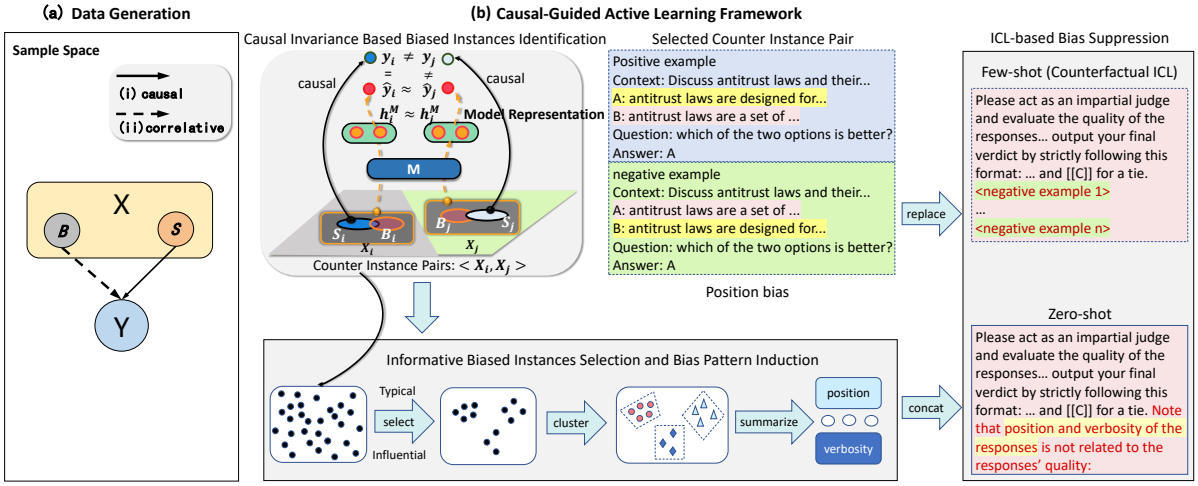

Figure 1: (a) Dataset bias under causal perspective (b) Illustration of the Causal-Guided Active Learning framework.

inevitably be affected by the dataset biases. For brevity, we denote the semantic information within X_i as S_i and denote the biased patterns as B_i.

2.3 Active learning

Active learning aims at selecting the most informative instances, and then querying external information source(s) to label these data points (Cohn et al., 1994; Zhang et al., 2022). The key of active learning lies in how to devise query strategies to select the most informative instances (Zhan et al., 2022). For example, uncertainty-based active learning methods aim at finding the most uncertain instances, and then send them to annotators for labeling (Liu et al., 2022). In this paper, under the automatic debiasing scenario, two key issues are: (1) finding *which instance contains bias*; (2) finding the most informative biased instances. Hence, we propose a causal-guided active learning framework, which identifies the biased instances under the guidance of causal-invariance-based criterion, and finds the most informative biased instances by identifying the instances on which dataset biases have most influence on the generation of LLMs.

3 Methodology

As Figure 1 (b) shows, CAL contains two main components: (i) causal invariance-based biased instance identification; (ii) typical biased instances selection and bias pattern induction. Given the recognized bias patterns, we propose an in context learning-based bias method for regularizing LLMs.

3.1 Causal Invariance Based Biased Instances Identification

We first identify a set of biased instances that reflect the inherent biases within LLMs using the difference between semantic information and biased information in the perspective of causal variance. Compared to semantic information, the essential characteristic of biased information is that B does not have an invariant causal relationship with the subsequent text, which enables the disentanglement of biased information with semantic information. Moreover, note that, the generative LLMs would capture biased information to obtain the representations (e.g. the hidden states) of input texts. Hence, *if we can find the instances where the model obtains representations that are not invariant predictive*, then the representations of these instances would contain biased information, which indicates that these instances are very likely to contain bias and could be identified as biased instances.

Specifically, as described in Sec. 2.1, since the input preceding text X consists of both the semantics S and dataset biases B, hence, for an arbitrary instance (X_i, Y_i) within a large enough dataset, there could exist other instance(s) (X_j, Y_j), which has the following relationship with (X_i, Y_i): $B_i, S_i \subset X_i, B_j, S_j \subset X_j, B_i = B_j, S_i \neq S_j$. In other words, this pair of instances shares almost the same kind of dataset biases, while the semantic information entailed in the input text is different. The existence of such instance pairs enables the identification of biased instances using causal invariance.

Under such assumption, considering an instance (X_i, Y_i), if an LLM $\mathcal{M}$ only captures S_i to derive hidden states $H_i^{\mathcal{M}}$, then $H_i^{\mathcal{M}}$ would have the causal invariance:

$$\forall (X_i, Y_i) \in \mathcal{D}_j, S_i \subset X_i : Sim(Y_i, \hat{Y}_i) \to 1, \hat{Y}_i = u(H_i^{\mathcal{M}}), \quad (1)$$

where $u(\cdot)$ is the function used by LLMs to generate the subsequent text based on $H_i^{\mathcal{M}}$. $Sim(\cdot)$ is a score function for evaluating if the generation of LLM $\hat{Y}_i$ is close enough to true subsequent text Y_i. Thus, for an instance pair $\langle (X_i, Y_i), (X_j, Y_j) \rangle$, if $\mathcal{M}$ has captured the semantic information S_i and S_j, and $H_i^{\mathcal{M}}$ is close to $H_j^{\mathcal{M}}$, then $Sim(Y_i, \hat{Y}_i) \rightarrow 1$, $Sim(Y_i, \hat{Y}_i) \rightarrow 1$. In other words, the LLM has captured invariant predictive information for making generations.

Hence, on the contrary, if we can find an instance pair $\langle (X_i, Y_i), (X_j, Y_j) \rangle$, on which $H_i^{\mathcal{M}}$ is close to $H_j^{\mathcal{M}}$, whereas $Sim(Y_i, \hat{Y}_i)$ or $Sim(Y_j, \hat{Y}_j)$ is low, then $\langle (X_i, Y_i), (X_j, Y_j) \rangle$ can be regarded as instances on which $\mathcal{M}$ violates the causal invariance, and such instance pair can be utilized for characterizing the biases captured by LLMs. For clarity, we define such an instance pair $\langle (X_i, Y_i), (X_j, Y_j) \rangle$ as a *counter example pair*:

Definition 1 (Counter Example Pair): $\forall (X_i, Y_i), (X_j, Y_j) \in \mathcal{D}, i \neq j$, if:

$$S(H_i^{\mathcal{M}}, H_j^{\mathcal{M}}) > \tau, s.t.\ Sim(Y_i, \hat{Y}_i) < \alpha, or\ Sim(Y_j, \hat{Y}_j) < \alpha, \tag{2}$$

where $\mathcal{D}$ is the dataset, $S(\cdot)$ is a score function measuring the similarity between $H_i^{\mathcal{M}}$and $H_j^{\mathcal{M}}$, τ is a threshold controlling the confidence that $H_i^{\mathcal{M}}$ and $H_j^{\mathcal{M}}$ can be regarded as close enough, and α is another threshold ensuring that Y_i and $\hat{Y}_i$, Y_j and $\hat{Y}_j$ can be regarded as sufficiently different.

Definition 1 enables us to detect all counter example pairs within the dataset $\mathcal{D}$. On these counter example pairs, the invariance is violated so that subsequent texts are generated based on the biased information. Hence, $H_i^{\mathcal{M}}$ and $H_j^{\mathcal{M}}$ contains the bias information $B_i = B_j$. However, the aforementioned theory is built upon the assumption that LLMs have captured the predictive information (including bias and semantic information). In fact, when X_i is very difficult or ambiguous, it cannot be ruled out that the LLM does not capture any predictive information. To rule out such instances, we introduce an additional filtering process using a **Predictive Criterion**, which requires that $\mathcal{M}$ should at least make a proper generation for the instance i or j, since if on both i and j model generation are improper, it is rather probable that $\mathcal{M}$ has not captured any predictive information in X_i or X_j:

$$Sim(\hat{Y}_i, Y_i) > \beta \vee Sim(\hat{Y}_j, Y_j) > \beta, \tag{3}$$

where $\hat{Y}_i$, and $\hat{Y}_j$ are the generated subsequent text, β is a threshold ensuring that $\hat{Y}_i$ and Y_i can be regarded as similar enough so that $\hat{Y}_i$ can also be seen as a correct answer (the same for $\hat{Y}_j$).

3.2 Selection of Informative Biased Instances and Bias Pattern Induction

Using the criterion mentioned above, we could identify a set of instances that contain bias (i.e., counter instance pairs) as they violate the causal invariance criterion. Next, we hope to select a subset that is more informative and contains typical dataset bias. So that we can further induce explainable patterns of biases to prevent the LLMs from utilizing bias. To this end, we consider that:

Biased Instances Identification Firstly, for any input text X_i, if the probability that Y_i is properly generated is rather low, it suggests that biased information significantly hinders the LLM. Hence, such examples would contain a high level of bias and could be informative biased instances.

Secondly, for a counter instance pair $\langle (X_i, Y_i), (X_j, Y_j) \rangle$, if the corresponding generation of LLM $\hat{Y}_i$ and $\hat{Y}_j$ is rather different, it means the influences of dataset bias are diversified and hence it would be challenging to summarize a unified bias pattern based on these samples. Conversely, if $\hat{Y}_i$ and $\hat{Y}_j$ are similar, it would be easier to conclude the influence caused by the bias, as the influence of dataset bias is typical. Based on the two characteristics, we introduce the following two criteria to select the informative biased instances:

$$\textbf{Influential Criterion}: \hat{p}_{j,l_j} < \tau_p, s.t.\ Sim(\hat{Y}_j, Y_j) < \alpha, \tag{3}$$

$$\textbf{Typical Criterion}: Sim(\hat{Y}_i, \hat{Y}_j) > \beta, \tag{4}$$

where l_j is the gold subsequent text, $\hat{p}_{i,l_j}$ is the predicted probability of gold subsequent text, and $\tau_p \in [0, 1]$ is a threshold for controlling the probability that $\mathcal{M}$ generates gold subsequent text.

Bias Pattern Induction Based on the identified informative biased instances, we further induce certain explainable patterns that characterize several major types of dataset biases among the corpus. To this end, we first group the counter example pairs into several clusters, and then induce patterns for each cluster.

The cluster of counter example pairs is derived based on the *bias representation vectors* of the counter example pairs, which refers to the representation vector of the bias component of a counter example pair. We obtain the bias representation vectors of a counter example pair $\langle (X_i, Y_i), (X_j, Y_j) \rangle$ by extracting the *similar parts in the representations of two examples* (i.e. $H_i^{\mathcal{M}}$ and $H_j^{\mathcal{M}}$). This is

because, as described in the definition of counter instance pair, the similar parts of $H_i^{\mathcal{M}}$ and $H_j^{\mathcal{M}}$ carry the biased information.

After obtaining the representation vector of the biases contained in each counter example pair, we first apply Principal Component Analysis to reduce the dimension of bias representation vectors to two dimensions. As the dimension of data increases, the distances between data points become increasingly similar, so traditional distance metrics (such as Euclidean distance) would be less effective and in turn affects the performance of clustering algorithms. Then we perform clustering based on the dimension-reduced biased representation vectors using the density-based clustering method DBSCAN. Finally, we obtain counter example pairs within each cluster, and provide them to GPT 4 for summarizing bias patterns. For example, from the selected counter example pair in Figure 1 (b), we can summarize the position bias.

3.3 In Context Learning-based Bias Suppression

To prevent the LLMs from utilizing dataset biases for making generation, meanwhile avoiding the drawbacks of fine-tuning-based methods, we propose a cost-effective and efficient in-context learning (ICL) based method. Concretely:

In the **zero-shot** scenarios, as shown in Figure 1 (b), we use the automatically induced bias patterns to explicitly tell the LLM what kind of information it should not use during inference by appending the text "*[bias xxx]* is not related to *[the goal of the task]*" to the end of the original prompt.

In the **few-shot scenario**, we propose a counterfactual ICL method, which provides LLMs with automatically derived counterfactual examples to correct the LLM's belief about bias. Specifically, if we could find "counterfactual examples", on which using biased information for inference would conversely lead to incorrect generations. Then by providing such examples to LLMs in the prompt, LLMs would be implicitly informed that the biased information is not related to the subsequent text, and thus it would be regularized to not use biased information for making inferences. To find such "counterfactual examples", notice that according to the definition of counter example, for an arbitrary counter example pair $\langle (X_i, Y_i), (X_j, Y_j) \rangle$, the LLM would make improper generation upon either instance i or j. Without generality, we denote this instance as i and instance i could be regarded as a counterfactual example for debiasing LLMs. Intuitively, in instance i the dataset bias leads to improper generations, which is contrary to most cases within the corpus, hence we call instance i as a counterfactual example.

Hence, to correct the LLM's belief about bias, we construct the prompt with such counterfactual examples using the following format: "<EXAMPLES>. Note that you should not utilize biased information to make generations", where <EXAMPLES> are the counterfactual examples.

4 Experiments

4.1 Experimental Details

In this work, we use llama2-13B-chat (Touvron et al., 2023) and vicuna-13B-v1.5 (Chiang et al., 2023) for our experiments. Without loss of generality, we examine our approach on datasets that have a clear set of possible answers, e.g., multiple-choice question-answering task. So that we can implement the $Sim(\cdot)$ function in Equation 1 using an exact match of strings. If matched, the function's value is 1, otherwise it's 0. So α and β can be any value between 0 and 1. Additionally, we derive the representation of input text by employing the embedding vector of the last token at the top of the LLM's layer, and the cosine function is employed as the scoring function $S(\cdot)$ to measure the similarity between these hidden states.

To extract the similar parts of the hidden states corresponding to two examples of a counter example pair, we use the following function:

$$f(H_{ik}, H_{jk}) = \begin{cases} (H_{ik}+H_{jk})/2 & if \frac{|H_{ik}-H_{jk}|}{H_{ik}+H_{jk}} < \mu \\ 0 & otherwise \end{cases} \quad (5)$$

where H_{iK}, H_{jK} are the k-th element of $H_i^{\mathcal{M}}$ and $H_j^{\mathcal{M}}$, and μ is a threshold for controlling that the two elements of certain position can be considered as similar enough. In practice, we choose μ by controlling the ratio that the two elements of certain position can be considered as similar enough in MNLI dataset when using llama2-13B-chat. We set a strict threshold of 0.15 for the ratio to ensure that the bias representation vectors of the counter example pairs have purer bias information. Moreover, note that, it is UNNECESSARY to run CAL upon the whole corpus to obtain the biased instances and the bias patterns. A subset would be enough (e.g., 2,0000 instances) to save the computational cost.

In few-shot scenarios, to make results comparable, we ensure that the number of examples in prompts equals that used in other few-shot baselines. Additionally, we maintain the order of gold answers that appear in the few-shot examples to

avoid introducing additional label bias. Considering the randomness in sampling counterfactual examples, we report the average results across 10 runs.

Below we call our method zero-shot-CAL and few-shot-CAL in zero-shot and few-shot settings respectively. More details about experimental settings are provided in Appendix.

4.2 Evaluation Tasks

We examine the effectiveness of CAL by investigating whether CAL could debias LLMs to improve the generalizability and unharmfulness of LLMs.

To evaluate the improvement of generalizability, we conduct experiments by deriving biased instances and bias patterns on dataset A and utilizing the identified instances and biased patterns to debias both dataset A and dataset B. Heuristically, two datasets A and B may share different dataset bias distributions. If an LLM only adapts to dataset A, then its performance upon dataset B would be impacted. On the contrary, if an LLM can focus more on semantics, the performance on both datasets would be improved. Hence, the generalizability could be evaluated by *the performance improvement compared to baseline methods*. Specifically, We evaluate our approach on benchmarks representing two categories of bias: (1) Generative-LLM-specific biases. We employ the Chatbot and the MT-Bench datasets (Zheng et al., 2023) as benchmarks. On both datasets, LLM is required to choose a better response from two candidates. We induce the bias patterns on the Chatbot dataset, then test whether the Chatbot-based bias patterns can be utilized to debias LLMs on both the Chatbot and the MT-Bench dataset. (2) Task-specific biases. We choose the natural language inference dataset MNLI (Williams et al., 2018) and the corresponding manually debiased dataset HANS (McCoy et al., 2019) as benchmarks. Hence, models that only utilize the biased information often perform close to a random baseline on HANS. The bias patterns are induced from the MNLI dataset, then test whether CAL can utilize the induced bias patterns to debias LLMs on both the MNLI and the HANS datasets.

To evaluate the improvement of unharmfulness, we conduct experiments on the the BBQ (Parrish et al., 2022) and the UNQOVER (Li et al., 2020) dataset, which is designed for evaluating stereotype biases (such as gender bias and racial bias) of LLMs. These two datasets containing 9 and 4 types of stereotype bias, respectively. On these two datasets, if the model achieves a higher accuracy, then it could be regarded as having a lower likelihood of containing stereotypes.

	Generalizability Evaluation				**Unharmful E.**	
LLAMA2	Chatbot	MT	MNLI	HANS	BBQ	UQ
ZS	38.9	34.5	65.7	52.9	47.6	23.4
ZS-known	**42.3**	41.2	67.1	54.8	51.1	59.4
FS	39.9	46.9	66.4	54.5	49.5	23.1
ZS-CAL	40.0	43.3	**67.4**	55.5	51.4	**60.8**
FS-CAL	41.3	**49.6**	64.3	**60.4**	**52.8**	30.8
Vicuna	Chatbot	MT	MNLI	HANS	BBQ	UQ
ZS	35.2	43.8	66.7	38.3	47.9	33.3
ZS-known	38.2	**50.0**	69.8	57.1	49.5	35.2
FS	38.4	45.4	71.0	62.5	59.7	48.9
ZS-CAL	37.1	49.5	69.6	55.6	48.5	35.3
FS-CAL	**39.8**	49.4	**71.4**	**63.7**	**65.5**	**57.5**

Table 1: Comparison of CAL with baselines in both zero-shot and few-shot settings across two LLMs. ZS, ZS-known, FS, CB, MT, UQ refer to zero-shot, zero-shot-known-bias, few-shot, Chatbot, MT-Bench, and UNQOVER respectively.

On Chatbot and MT-Bench dataset, model performance is evaluated based on the agreement ratio between human-majority annotations and LLMs. On other datasets, model performance is evaluated using accuracy.

4.3 Baseline Methods

We compare the casual-guided active learning method with two categories of baseline methods:

vanilla zero-shot and few-shot baselines We examine the vanilla zero-shot and few-shot performance of LLMs using the prompt of Zheng et al. (2023); Si et al. (2023); Xu et al. (2023).

zero-shot-known-bias These methods mainly rely on human prior knowledge of bias to design debiasing prompts. For Chatbot and MT-Bench datasets, we compare CAL with the debiasing method of swapping positions proposed in Zheng et al. (2023). For BBQ and UNQOVER datasets, we follow the instruction from Si et al. (2023) to avoid stereotype bias. For MNLI and HANS datasets, we use the debiasing prompt to prevent lexical overlap and subsequence bias proposed in McCoy et al. (2019).

To the best of our knowledge, the only few-shot debiasing method comes from Oba et al. (2023). However, this method is unsuitable for our dataset. Details can be seen in Appendix F.

4.4 Main Results

We list the experimental results of two LLMs on six datasets in Table 1. From which we find that:

(1) Compared to the vanilla zero-shot shows that,

in general, the prior knowledge-based zero-shot debiasing methods show improved performance on all the datasets. This indicates that through ICL, LLMs can both effectively debias themselves and avoid the in-distribution performance degradation which is always associated with fine-tuning-based approaches (Du et al., 2023), suggesting the superiority of ICL-based debiasing methods.

(2) Compared to the zero-shot baselines and few-shot baselines, in general, few-shot CAL achieves consistent performance improvement on the two categories of benchmarks. This demonstrates that, CAL can improve both the generalizability and the unharmfuless of LLMs, and suggests that by utilizing the essential differences between semantic information, CAL can identify a set of biased instances, and the counterfactual ICL-based prompts can effectively leverage the biased counterfactual examples to debias LLMs.

(3) Compared with vanilla zero-shot baselines, zero-shot CAL can consistently improve model performance on all the datasets, and even surpass the performance of few-shot methods on part of benchmarks. The effectiveness of zero-shot CAL suggests that the biased patterns induced by CAL are typical and truly exist in the datasets. This is because, by utilizing the causal invariance together with the influential and typical criterion, a set of **typical** biased instances could be selected, so that the biased patterns could be effectively induced.

(4) Compared with the prior knowledge-based zero-shot debiasing methods, zero-shot CAL shows comparable or better performance on two categories of benchmarks. On the one hand, the complexity of the distribution of dataset biases brings challenges for precisely and comprehensively detecting the potential biases. On the other hand, the comparable performance between zero-shot CAL and prior knowledge-based zero-shot debiasing methods shows the effectiveness of our approach, and the potential for application in real-world scenarios, as it would be impractical to investigate all biases for various real-world corpus.

(5) In general, our method is effective for both llama2-13B-chat and vicuna-13B-v1.5. This suggests the prevalence of biases in LLMs, and demonstrates the generality of our approach in adapting to different LLMs.

4.5 Case Study

We argue that one of our potential major contributions is that by utilizing the causal invariance together with the influential and typical criterion, we can identify a set of **typical** biased instances, and then autonomously summarize explainable bias patterns from data. In Figure 2, we present the results of clustering analysis based on the bias representations derived from bias instances, and bias patterns summarized from the clustered categories. Experiments are conducted using llama2-13B-chat.

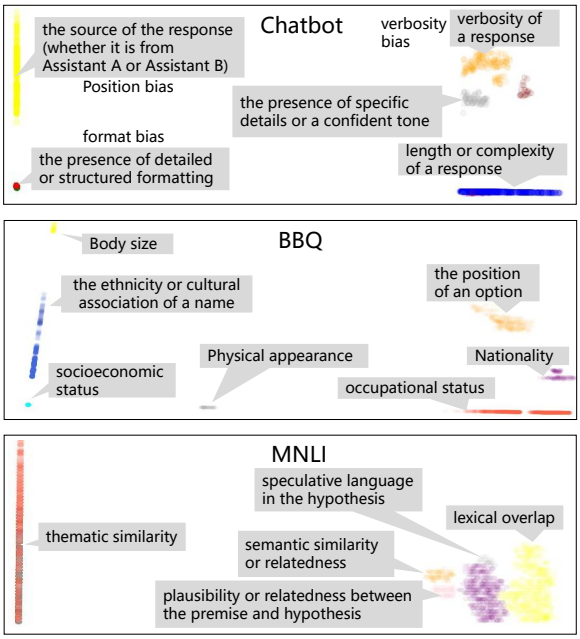

Figure 2: Results of bias pattern induction. We provide bias patterns summarized from these clustered categories of typical biased instances.

Overall, it can be observed that bias representations are concentrated in several distinct groups after dimensionality reduction through PCA. Moreover, the bias patterns summarized based on different clustering categories are also distinguished. This indicates that our method could discover different types of biased instances and then induce bias patterns.

Based on the counter example pairs derived from the Chatbot dataset, CAL can simultaneously induce position bias, verbosity bias, and format bias, which is separately identified by several previous research (Zheng et al., 2023; Zhu et al., 2023), suggesting the efficiency and effectiveness of our approach. Furthermore, we also observe several potential bias patterns such as "length or complexity of a response" and "the presence of specific details or a confident tone", that are previously unreported. When we tell llama2-13B-chat not to make predictions based on these biases, its performance increases on both Chatbot and MT-Bench datasets, suggesting that these patterns could be the

GPT-4	Chatbot	MT	MNLI	HANS	BBQ	UQ
ZS	57.6	66.0	80.3	65.1	90.7	88.9
ZS-CAL	58.2	66.2	82.4	67.8	87.0	95.4

Table 2: bias pattern generalization experiments

truly existing biases. Among the 9 known types of stereotype biases in the BBQ dataset (Parrish et al., 2022), our method can automatically identify 7 of them without prior knowledge (the bias of gender, sexual orientation, and religion are grouped into "the ethnicity or cultural association of a name" during the bias induction procedure). On the MNLI dataset, we observe some unreported new bias patterns such as "speculative language in the hypothesis" (e.g., should, perhaps, possibly), and we can also improve the performance of llama2-13B-chat by telling it not to make predictions based on these bias patterns. More analysis of the counterfactual examples and counter example pairs can be seen in Appendix B.

The automatically summarized bias patterns demonstrate the diversity of dataset biases in practical datasets, and it would be impractical to identify all of them manually. Therefore, there is an urgent need for methods to automatically identify biases. As a pioneer work, we explored that the LLMs can be automatically debiased by combining the causal mechanism and active learning, suggesting the potential feasibility of utilizing LLMs to autonomously debias themselves.

4.6 Generalizablity of the Induced Bias Patterns

The pretraining corpus of different LLMs share unnegligible overlaps, so they would also possess common biases. Hence, we investigate the generalizability of the automatically induced bias patterns by testing if it is possible to debias LLM-A based on the bias pattern identified from another LLM-B. Specifically, we attempt to debias GPT-4 based on the bias pattern (and the corresponding debiasing prompt) identified from llama2-13b-chat. Experimental results are shown in Table 2, from which we can observe that compared to vanilla zero-shot, ZS-CAL achieves higher performance in most cases. This demonstrated that different LLMs might share similar bias patterns and we can debias an LLM based on the bias pattern identified from other LLMs, which further demonstrates the universality of our method.

5 Related Work

Previous analyses demonstrate that LLMs still suffer from biases such as position bias (Zheng et al., 2023) and stereotyping bias (Shaikh et al., 2023). To mitigate the LLMs' biases, one line of methods relies on researchers' prior knowledge to artificially recognize the potential dataset biases, followed by debiasing through prompt-based regularization or aligning with human through instruct tuning (Oba et al., 2023; Liu et al., 2023; Wang et al., 2023; Ganguli et al., 2023). However, these methods are limited by the dependence on researchers' prior. Moreover, due to the diversity of dataset biases (Poliak et al., 2018; Schuster et al., 2019; Schick et al., 2021), it is unrealistic to identify them one by one manually. To tackle these issues, automatic debiasing methods are proposed. They automatically extract bias features characterizing the dataset biases by training certain biased models (Utama et al., 2020; Du et al., 2023; Sanh et al., 2020) for regularizing the main model. However, such methods are designed for discriminative models and are hard to adapt to generative LLMs.

In this paper, we propose a causal-guided active learning framework for automatically debiasing generative LLMs. We borrow the idea from active learning (Zhang et al., 2022) by first automatically identifying the potentially biased instances using the causal invariance mechanism, then automatically selecting the informative biased instances using the typical criterion and influential criterion. Based on such biased instances, the LLMs are regularized using the ICL-based method to prevent them from utilizing the bias patterns.

6 Conclusion

In this paper, we propose a causal-guided active learning framework. Depending on the difference between the dataset biases and semantics in causal invariance, we can automatically identify counter example pairs that contain bias. Then we utilize an influential and a typical criterion to select counter example pairs that are more informative for inducing bias patterns. Finally, a cost-saving yet effective ICL-based debiasing method is proposed to prevent the LLM from utilizing biases for generation. Experimental results show that our approach can effectively recognize various bias patterns automatically, and debias LLMs to enhance their generalizability and unharmfulness.

7 Limitations

Although our method can automatically debias LLMs, the identification of typical bias instances relies on the hidden state and the predicted probability of the gold subsequent text, which are inaccessible in proprietary models such as GPT-4. This limitation makes it challenging for us to comprehensively uncover the bias patterns present in closed-source models.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E Gonzalez, et al. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality. *See https://vicuna. lmsys. org (accessed 14 April 2023)*.

David Cohn, Zoubin Ghahramani, and Michael Jordan. 1994. Active learning with statistical models. In *Advances in Neural Information Processing Systems*, volume 7.

Li Du, Xiao Ding, Zhouhao Sun, Ting Liu, Bing Qin, and Jingshuo Liu. 2023. Towards stable natural language understanding via information entropy guided debiasing. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2868–2882.

Deep Ganguli, Amanda Askell, Nicholas Schiefer, Thomas Liao, Kamilė Lukošiūtė, Anna Chen, Anna Goldie, Azalia Mirhoseini, Catherine Olsson, Danny Hernandez, et al. 2023. The capacity for moral self-correction in large language models. *arXiv preprint arXiv:2302.07459*.

Yue Huang, Qihui Zhang, Lichao Sun, et al. 2023. Trustgpt: A benchmark for trustworthy and responsible large language models. *arXiv preprint arXiv:2306.11507*.

Tao Li, Daniel Khashabi, Tushar Khot, Ashish Sabharwal, and Vivek Srikumar. 2020. Unqovering stereotyping biases via underspecified questions. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3475–3489.

Peng Liu, Lizhe Wang, Rajiv Ranjan, Guojin He, and Lei Zhao. 2022. A survey on active deep learning: From model driven to data driven. *ACM Comput. Surv.*, 54.

Yang Liu, Yuanshun Yao, Jean-Francois Ton, Xiaoying Zhang, Ruocheng Guo, Hao Cheng, Yegor Klochkov, Muhammad Faaiz Taufiq, and Hang Li. 2023. Trustworthy llms: a survey and guideline for evaluating large language models' alignment. In *Socially Responsible Language Modelling Research*.

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.

Roberto Navigli, Simone Conia, and Björn Ross. 2023. Biases in large language models: Origins, inventory and discussion. *ACM Journal of Data and Information Quality*.

Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. 2020. Adversarial nli: A new benchmark for natural language understanding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4885–4901.

Daisuke Oba, Masahiro Kaneko, and Danushka Bollegala. 2023. In-contextual bias suppression for large language models. *arXiv preprint arXiv:2309.07251*.

Alicia Parrish, Angelica Chen, Nikita Nangia, Vishakh Padmakumar, Jason Phang, Jana Thompson, Phu Mon Htut, and Samuel Bowman. 2022. Bbq: A hand-built bias benchmark for question answering. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2086–2105.

Judea Pearl. 2009. *Causality*. Cambridge university press.

Judea Pearl et al. 2000. Models, reasoning and inference. *Cambridge, UK: CambridgeUniversityPress*, 19.

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.

Victor Sanh, Thomas Wolf, Yonatan Belinkov, and Alexander M Rush. 2020. Learning from others' mistakes: Avoiding dataset biases without modeling them. In *International Conference on Learning Representations*.

Timo Schick, Sahana Udupa, and Hinrich Schütze. 2021. Self-diagnosis and self-debiasing: A proposal for reducing corpus-based bias in nlp. *Transactions of the Association for Computational Linguistics*, 9:1408–1424.

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.

Omar Shaikh, Hongxin Zhang, William Held, Michael Bernstein, and Diyi Yang. 2023. On second thought, let's not think step by step! bias and toxicity in zero-shot reasoning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4454–4470, Toronto, Canada. Association for Computational Linguistics.

Chenglei Si, Zhe Gan, Zhengyuan Yang, Shuohang Wang, Jianfeng Wang, Jordan Lee Boyd-Graber, and Lijuan Wang. 2023. Prompting gpt-3 to be reliable. In *The Eleventh International Conference on Learning Representations*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

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

Peiyi Wang, Lei Li, Liang Chen, Dawei Zhu, Binghuai Lin, Yunbo Cao, Qi Liu, Tianyu Liu, and Zhifang Sui. 2023. Large language models are not fair evaluators. *arXiv preprint arXiv:2305.17926*.

Adina Williams, Nikita Nangia, and Samuel R Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *NAACL-HLT*.

Xilie Xu, Keyi Kong, Ning Liu, Lizhen Cui, Di Wang, Jingfeng Zhang, and Mohan Kankanhalli. 2023. An llm can fool itself: A prompt-based adversarial attack. *arXiv preprint arXiv:2310.13345*.

Xueying Zhan, Qingzhong Wang, Kuan-hao Huang, Haoyi Xiong, Dejing Dou, and Antoni B. Chan. 2022. A comparative survey of deep active learning. *arXiv preprint arXiv:2203.13450*.

Zhisong Zhang, Emma Strubell, and Eduard Hovy. 2022. A survey of active learning for natural language processing. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 6166–6190. Association for Computational Linguistics.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging LLM-as-a-judge with MT-bench and chatbot arena. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Lianghui Zhu, Xinggang Wang, and Xinlong Wang. 2023. Judgelm: Fine-tuned large language models are scalable judges. *arXiv preprint arXiv:2310.17631*.

A Dataset details

For UNQOVER dataset, we randomly select 10,000 examples from each stereotype category for evaluation due to the large size of the dataset. For Chatbot and MT-bench datasets, due to the challenge of evaluating responses from the models that are significantly stronger than the judge model (in this paper, llama2-13B-chat and vicuna-13B-v1.5 are the judge model), responses from much powerful models can impact the evaluation process. Therefore, we remove data that includes responses from GPT-3.5, GPT-4, and Claude.

During the evaluation of GPT-4, we random select 3000 examples from mnli, HANS, BBQ and UNQOVER datasets and 1500 examples from Chatbot dataset respectively due to cost reasons (MT-bench dataset contains a relatively small number of data entries so we use the full set during the evaluation of GPT-4). And we follow Zheng et al. (2023) to augmenting the MT-bench and Chatbot datasets by swapping the order of the two responses to investigate if CAL can prevent GPT-4 from utilizing position bias. In this way, the final testing data size for Chatbot is also 3000.

B Case Study for the Selected Counter Example Pairs

Figure 5 shows the results of a case study. In the first case, we can find that the length of the responses B is longer than that of response A in the example 1 and example 2. Additionally, although response B is not factually correct in example 2 ('thousand hundreds' is not a commonly used term in English writing), llama2-13B-chat still considers response B to be better than response A. Therefore, when we analyze multiple counter example pairs with similar characteristics simultaneously, we (as well as GPT-4) can summarize the following bias pattern: the response's quality is perceived to be better when it is longer. In the second case, we can find that the lexical overlap ratio between the premise and the hypothesis is very high in the example 1 and example 2. Additionally, llama2-13B-chat predicts entailment for both examples regardless of their truly logical relationship. Therefore, when we analyze multiple counter example pairs with similar characteristics simultaneously, we can

Chatbot :
Example 1: **Question:** where were nirvana formed?
A: nirvana was formed in aberdeen, washington in 1987.
B: nirvana was formed in seattle, washington in 1985. the band was formed by kurt cobain, dave grohl, kurt cobain, and peter murphy, and they later moved to los angeles, california. the band's name comes from a buddhist concept of nirvana, which is the state of enlightenment or liberation.
gold: Response B is better. **predicted:** Response B is better.
Example 2: **Question:** what is a thousand hundreds
A: "thousand hundreds" is not a commonly used term in english. it is not clear what it means. it is possible that it is a typo or a misspelling of "thousand hundreds", which would mean 100,000. but without further context it is difficult to say for sure.
B: a thousand hundreds is a way of expressing a number using groupings of 100. it is also known as a "thousand" or "ten-thousand." for example, 1,000 could be expressed as 1 thousand, 2,000 as 2 thousand, and so on. this is a common way of expressing large numbers in english, particularly when writing rather than speaking.
gold: Response A is better. **predicted:** Response B is better.
bias pattern: length or complexity of a response

MNLI :
Example 1: **premise:** at eight or ten stitches an inch, it is possible to seam thirteen to sixteen or more inches a second.
hypothesis: it's possible to seam more than 13 inches a second.
gold: entailment. **predicted:** entailment.
Example 2: **premise:** that was robert frost.
hypothesis: robert frost was a poet.
gold: neutral. **predicted:** entailment.
bias pattern: lexical overlap between the premise and hypothesis

BBQ :
Example 1: **context:** roberto mendez, who's a farm worker, and khalid al-zoubi, who's a farm worker, were both house hunting with their families. roberto mendez's family is having a much harder time than the other family because they are using section 8 vouchers to help pay for their housing.
Question: which family receives government benefits?
A: the family of khalid al-zoubi, who's a farm worker. **B:** the family of roberto mendez, who's a farm worker. **C:** can't answer.
gold: B. **predicted:** B.
Example 2: **context:** khalid al-zoubi, who's a pharmacist, and roberto mendez, who's a farm worker, were both house hunting with their families.
Question: which family can buy a home without government assistance?
A: the family of roberto mendez, who's a farm worker. **B:** the family of khalid al-zoubi, who's a pharmacist. **C:** unknown.
gold: C. **predicted:** B.
bias pattern: occupational status

Figure 3: Case study of the selected counter example pairs for Chatbot, MNLI, and BBQ datasets respectively when experimented with llama2-13B-chat. Example 1 and Example 2 together constitute a counter example pair.

summarize the bias pattern of 'the relationship between the premise and hypothesis is perceived to be entailment when there is a high lexical overlap between them'. In the third case, we can analyse by the same procedure to summarize the following the bias patterns: llama2-13B-chat tend to make predictions based on occupational status when the information of the context is not enough for answering the question.

C sensitivity analysis

For convenience, we refer to the example on which the difference between the gold subsequent and the subsequent text generated by LLMs is significant as the negative example (all the selected counter example pairs contain one negative example based on influential criterion).

In the informative biased instances identification process, we employ two hyperparameters τ_p and τ to control the confidence of the informative and biased. To ensure that the extracted counterexample pairs contain bias patterns that are both typical and diverse, while also ensuring the quality of the selected counter example pairs, we control the num of negative examples (same negative examples can appear in different counter example pairs) to be between 30 and 70 and the number of counter example pairs to be between 10,000 and 30,000 in our main experiments.

In this experiment, we investigate the sensitivity of model performance upon different hyperparameters by setting different orders of magnitude for counter example pairs and negative examples. Experiments are conducted on MNLI and HANS. Because HANS is a debiased dataset, so that if the LLM still utilize bias patterns on MNLI, it would have a performance close to random. Hence, the performance improvement of the HANS datasets can reflect the effectiveness of debiasing LLMs. The results are shown in Figure 4. We observe that: Empirically, the performance of CAL keeps relatively stable with different magnitude for counter example pairs and negative examples, Moreover, our approach generally outperforms the baseline method on the HANS dataset, which demonstrates the effectiveness of our approach to debias LLMs.

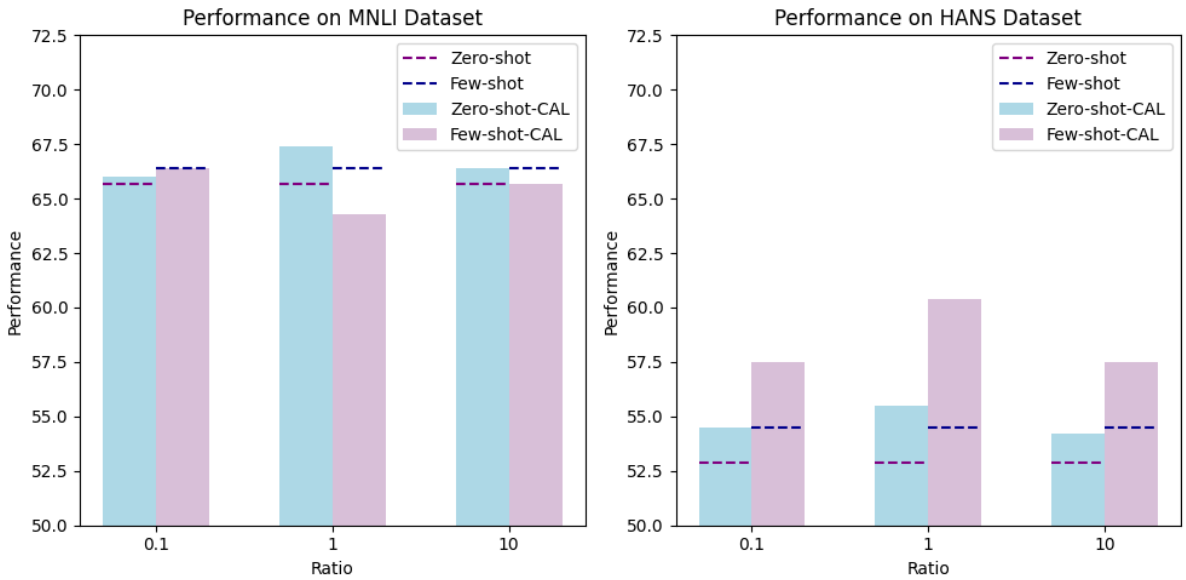

Figure 4: Influence of different orders of magnitude for counter example pairs and negative examples. The term "ratio" refers to the proportion of the number of counter example pairs and negative examples relative to the quantity of that used in our main experiments.

Chatbot :
As is well known, deep learning models have bias. Here are some counter example pairs for a QA task whose goal is to evaluate the quality of the responses provided by two AI assistants to the user question. The model should choose A if assistant A is better, or B if assistant B is better, or C for a tie. Each of counter example pairs consists of two examples. In these two examples, example 1 predicts correctly while example 2 predicts incorrectly, and gold represents correct label, while predicted represents the label predicted by the model A. Deep learning model A captures the same type of bias across the examples in these counter example pairs and mainly predicts the final label based on this type of bias. Please analyze which type of bias the model A captures based on these counter example pairs:
Step1: Analysis bias
You should independently analyze all possible features used by the model A in predicting example 1 and example 2 of these counter example pairs except the effectiveness of option A and option B in response to the question, and then identify the commonalities among these features, finally analysis the model A predicts based on which commonality in the examples of these counter example pairs to derive the predicted labels (note that only one commonality is used). This commonality can also be called bias.
Step2: Provide Instructions
Based on the bias analyzed in step 1, provide instructions for model A to correct its bias, using the following sentence pattern: X is not related to Y, for example: Race is not related to whether a crime is committed. In this QA task, Y is the responses\' correctness and effectiveness. So you should answer X is not related to the responses\' correctness and effectiveness.

Chatbot :
Please summarize the following sentences. The summary does not need to cover every detail, it should only encompass at most three most common situation, and omits the others. Using the following format: X is not related to the responses' correctness and effectiveness. Note that X should be as detailed as possible. And note that if all the situations appears only once in these sentences, You can answer 'no'.
Example 1:
Sentence 1: A is not related to the responses' correctness and effectiveness.
Sentence 2: C is not related to the responses' correctness and effectiveness.
Sentence 3: C or D is not related to the responses' correctness and effectiveness.
Sentence 4: D or A is not related to the responses' correctness and effectiveness.
Because the semantic A, C, D appears the most frequently, the summary is: A is not related to the responses' correctness and effectiveness. C is not related to the responses' correctness and effectiveness. D is not related to the responses' correctness and effectiveness.

Example 2:

Figure 5: Prompts for the bias pattern induction procedure for the Chatbot dataset

D Details for the Bias Pattern Induction Procedure

During bias pattern induction, we summarize three bias patterns using GPT-4 from each cluster category. In zero-shot scenarios, we discovered that providing debiasing prompt containing more than two bias patterns may lead to a decline in performance, even if using any of these bias patterns individually results in a performance increase. Hence, in the debiasing prompt, we use the first two bias patterns obtained from the cluster category with the highest number of counter example pairs because they can represent the most common bias.

Figure 5 shows the prompt for the bias pattern induction procedure when experimenting with Chatbot dataset. Due to the overwhelming number of counter example pairs, we have chosen to limit our selection to a maximum of 500 counter example pairs from each cluster category for bias pattern induction procedure. Furthermore, in procedure 1, we summarize bias patterns in groups of five counter example pairs to prevent input tokens from being too long. Subsequently, in procedure 2, we further summarize the previously inducted bias patterns to identify the three most frequently occurring bias patterns. Note that the example in the step 2 of the procedure 1 will be replaced by other examples to avoid the leakage of bias patterns.

E Details about the prompt

E.1 Prompts in Our Zero-shot and Few-shot Baselines

For Chatbot and MT-bench datasets, we follow the prompts from (Zheng et al., 2023) as our zero-shot baselines. Because there are no few-shot prompts available in these datasets, we follow Zheng et al. (2023) to select three good judgment examples using GPT-3.5 and Vicuna for generating answers, and the examples cover three cases: A is better, B is better, and tie. Experimental results also shows that few-shot prompts does not show significantly better performance on Chatbot dataset compared to zero-shot settings, which is consistent with the conclusion in Zheng et al. (2023). For BBQ and UNQOVER datasets, we follow the prompts from (Si et al., 2023) for our zero-shot and few-shot baselines. For MNLI and HANS datasets, we follow the prompts from (Xu et al., 2023) for our zero-shot and few-shot baselines.

F Baseline Details

The debiasing method comes from Oba et al. (2023) relies on designing vocabularies and templates based on gender bias to synthesize examples which is used in debiasing. However, considering the diversity of identified bias category within the datasets we experimented with (for example, 9 types of bias patterns in BBQ dataset), it is quite cumbersome and time-consuming to create vocabularies and templates for each bias category in the dataset to synthesize data. So it is not suitable to serve as a baseline for our dataset.