Exploring and Mitigating Shortcut Learning for Generative Large Language Models

Zechen Sun*, Yisheng Xiao*, Juntao Li†, Yixin Ji, Wenliang Chen, Min Zhang
Institute of Computer Science and Technology, Soochow University
Suzhou, China
{ysxiaoo,zcsuns}@stu.suda.edu.cn, jiyixin169@gmail.com
{ljt,wlchen,minzhang}@suda.edu.cn

Abstract

Recent generative large language models (LLMs) have exhibited incredible instruction-following capabilities while keeping strong task completion ability, even without task-specific fine-tuning. Some works attribute this to the bonus of the new scaling law, in which the continuous improvement of model capacity yields emergent capabilities, e.g., reasoning and universal generalization. However, we point out that recent LLMs still show shortcut learning behavior, where the models tend to exploit spurious correlations between non-robust features and labels for prediction, which might lead to overestimating model capabilities. LLMs memorize more complex spurious correlations (i.e., task $\leftrightarrow$ feature $\leftrightarrow$ label) compared with that learned from previous pre-training and task-specific fine-tuning paradigm (i.e., feature $\leftrightarrow$ label). Based on our findings, we propose FSLI, a framework for encouraging LLMs to **F**orget **S**purious correlations and **L**earn from **I**n-context information. Experiments on three tasks show that FSFI can effectively mitigate shortcut learning. Besides, we argue not to overestimate the capabilities of LLMs and conduct evaluations in more challenging and complete test scenarios.

Keywords: shortcut learning, spurious correlation, natural language understanding, large language model

1. Introduction

Pre-trained models (PLMs) have achieved promising performance in various tasks in the past few years (Devlin et al., 2018; Lewis et al., 2020; Raffel et al., 2020). Recently, as the computing resources increase, researchers gradually scale the model size or data size of original PLMs for superior performance (Shanahan, 2022; Hoffmann et al., 2022), known as large language models (LLMs), e.g., GLM (Zeng et al., 2022), LLaMA (Touvron et al., 2023) and GPT-4 (OpenAI, 2023). These LLMs can consistently achieve significant performance improvements and exhibit several special abilities (Wei et al., 2022) compared with original PLMs. For example, in-context learning is a brand-new skill (Brown et al., 2020), where LLMs can learn helpful information from task demonstrations with only a few input-output pairs concatenated. Moreover, the advanced training technologies such as instruction tuning (Wei et al., 2021) and reinforcement learning with human feedback (RLHF) (Ouyang et al., 2022) further equip the LLMs with the ability of zero-shot learning. Since these skills allow LLMs to handle different tasks without any additional training process or gradient update, which significantly enhances their generalization capability and usability. Surprisingly, the most representative LLM, GPT-4, has been proven to be able to benefit more than 1,800 different tasks (Bubeck et al., 2023). The development of LLMs goes further to our purpose of artificial general intelligence (AGI).

Despite the remarkable performance of recent LLMs, some challenges and problems still arise in real-world applications, such as hallucination, ethical and privacy concerns (Bang et al., 2023). Besides, many researchers probe the recent LLMs for some specific areas, and witness the degeneration in performance, e.g., ambiguity modeling (Liu et al., 2023), negative knowledge learning (Chen et al., 2023), etc. As a result, it is worth further exploration of whether LLMs truly understand intrinsic semantics rather than the surface form of texts.

In this paper, we explore shortcut learning (Du et al., 2022), where the models tend to exploit superficial non-robust features (Ilyas et al., 2019) (e.g., lexical overlap and specific content words) instead of robust features (e.g., semantic understanding and reasoning) to make predictions. It seriously hurts the generalization and robustness of natural language models, leading to inferior performance when applied to broader applications or more challenging scenarios (Geirhos et al., 2020). We could achieve a recognized consensus conclusion from previous related works that a model trained with more balanced datasets, more parameters, and more advanced learning strategies can help to mitigate the shortcut learning behavior (Tu et al., 2020; Ross et al., 2022; Bubeck and Sellke, 2023), and these all play a crucial role behind the recent success of LLMs. However, there exist no related explorations of shortcut learning for recent LLMs. Nat-

Zechen Sun and Yisheng Xiao contributed equally. Juntao Li is the corresponding author.

LREC-COLING 2024, pages 6883–6893
20-25 May, 2024.

urally, we wonder: **(1)** ***Do recent LLMs (such as ChatGPT) have shortcut learning behaviors under zero/few-shot learning settings?*** if have: **(2)** ***When and why do shortcut learning behaviors occur?*** and **(3)** ***How to mitigate them for LLMs?***

To answer the questions, we conduct analytical experiments to explore shortcut learning for LLMs (Section 3). Our results show that LLMs still suffer from shortcut learning and attribute this to the spurious correlations learned in the instruction tuning or RLHF process(Section 3.2). Next, we try to remove specific elements of the correlations to encourage LLMs to forget the spurious correlations and learn useful task information through in-context learning (Section 4). However, promising performances are only achieved with numerous examples in the demonstration. Therefore, we further introduce two methods to provide enough helpful task information in the demonstration with relatively fewer examples to mitigate shortcut learning and improve the overall performance (Section 5). Experiments on three widely-used datasets of different tasks demonstrate the effectiveness of our methods. In summary, our work provides a new perspective for evaluating the performance of LLMs. Models can mask their lack of semantic understanding through shortcut learning, and will exhibit significant performance gaps in simple testing scenarios and complex real-world scenarios. Therefore, it is crucial not to exaggerate the performance of LLMs and to conduct more realistic and comprehensive testing on them.

2. Background

2.1. Shortcut Learning

Shortcut learning is known to hurt the generalization of language models and has been well explored in recent years (Du et al., 2022). Based on original PLMs, researchers aim to look for the origins of shortcut learning (Tu et al., 2020; Lai et al., 2021; Si et al., 2023) and propose mitigation solutions (Stacey et al., 2020; Utama et al., 2020; Ross et al., 2022; Yao et al., 2022). However, we notice several limitations among them: (1) most explorations are based on BERT-like models, only a few works mention the generative language models, (2) the parameters of their models are always less than 1B, leading to limitations of their methods and conclusions. Recently, Schwartz and Stanovsky question the basic procedure of large-scale pretraining and task-specific fine-tuning paradigm and suggest focusing on zero/few-shot learning instead. These motivate us to explore shortcut learning in more comprehensive scenarios.

Lexical-overlap Bias	
Premise	The judges supported the manager and the lawyers
Hypothesis	The lawyers supported the manager.
Gold label	*Non-entailment*
Prediction	*Entailment*
Single-word Bias	
Premise	No, indeed, said Cynthia
Hypothesis	Certainly not, said Cynthia
Gold label	*Entailment*
Prediction	*Contradiction*

Table 1: Examples of lexical-overlap bias and single-word bias in natural language inference task, a high rate of lexical-overlap between the premise and the hypothesis can be a strong indicator of *Entailment*, and a negation word can be a strong indicator of *Contradiction*.

See Table 1. We introduce two representative formats. **Lexical-overlap bias** occurs if it contains two evaluation sentences with overlapping words, e.g., natural language inference (McCoy et al., 2020), reading comprehension (Lai et al., 2021). The language models view the overlap of two sentences as a shortcut and then make predictions without understanding the internal semantics. **Single-word bias** means that every single-word feature correlation is spurious (Gururangan et al., 2018; Gardner et al., 2021), e.g., numbers, negation words, adverbs of degree, etc. The models could make correct predictions and perform well in simple testing scenarios through shortcut learning. However, due to spurious correlations rather than semantic understanding, the excellent performance fails in more challenging or real-world settings. Shortcut learning seriously affects the robustness and performance of language models and may mislead researchers in evaluating model powers, which is worth further exploration.

2.2. In-context Learning

In-context learning (ICL) is first mentioned in GPT-3 (Brown et al., 2020) and allows the LLMs to learn specific abilities to solve different tasks with only a few examples in the demonstration. Then, LLMs can perform well in various tasks without updating model parameters. After adopting the specific prompt templates, some corresponding samples will be concatenated before the test input to serve as the demonstration (Dong et al., 2022). Specifically, take the natural language inference task as an example. Given one test instance $I_i = (x_i, y_i)$, and k examples in the demonstration, the model predicts the label formatted as:

$$P(y_i|x_i) = P(y_i|I_1 \oplus I_2, ..., \oplus I_k \oplus x_i), \qquad (1)$$

where $\oplus$ denotes the concatenation operation, we omit the prompt template. If $k = 0$, there is no example in the demonstration, it changes to the zero-shot setting. LLMs can perform implicit learning and identify the correct concept learned in the pre-training process via in-context learning.

2.3. Instruction Tuning and RLHF

Instruction tuning is where a pre-trained LLM is fine-tuned on a collection of natural language instructions containing various tasks and datasets (Wei et al., 2021, 2023b). It combines the appealing aspects of the pre-training and fine-tuning paradigms with prompting (Sanh et al., 2021). After instruction tuning, LLMs can show superior ability in zero-shot learning and achieve promising performance in unseen tasks (Wei et al., 2021; Chung et al., 2022). Surprisingly, instruction tuning can be combined with other prompting methods to improve performance, such as in-context learning and chain-of-thought prompting. Reinforcement learning from human feedback (RLHF) (Christiano et al., 2017; Ouyang et al., 2022) is another skill to boost the capacity of LLMs, which aims to align the outputs of LLMs with human values. Specifically, RLHF adopts a reward model that is trained with the human feedback data to provide the alignment score and then trains the LLMs with reinforcement learning (RL) algorithms (e.g., Proximal Policy Optimization (PPO) (Schulman et al., 2017)). After RLHF, the alignment criteria of LLMs (e.g., helpfulness, honesty, and harmlessness) will be greatly improved (Zhao et al., 2023).

3. Shortcut Learning of LLMs

Previous works have explored the shortcut learning problem and proposed mitigation solutions based on original PLMs (Friedman et al., 2022; Wen et al., 2022; Joshi et al., 2022; Eisenstein, 2022). Recently, LLMs have attracted much attention in the NLP community. Despite the remarkable performance, whether they still have the shortcut learning behavior remains unknown. Next, we first verify that LLMs also learn shortcut behaviors after instruction tuning or RLHF processes. Then, we further analyze the potential reasons.

3.1. Study Settings

We use HANS (McCoy et al., 2020) consisting of pairs of premise and hypothesis sentences with labels *entailment/non-entailment* as our evaluation dataset for early experiments. Premise and hypothesis sentences in this dataset all contain word overlaps. For backbone LLMs, we adopt different pairs of models without and with instruction tuning or RLHF but containing comparable parameters, e.g., GPT-3 *davinci* (Brown et al., 2020) and ChatGPT[1], LLaMA (Touvron et al., 2023) and Alpaca (Taori et al., 2023), T5 (Raffel et al., 2020) and Flan-T5 (Wei et al., 2021). Due to the limitation of computing resources, we randomly select 1,000 examples from the original development set to conduct experiments, and the rest are used for in-context learning. More specifically, we keep the same proportion of different labels, i.e., the numbers for examples with label *Entailment* and *Non-entailment* are both 500. We adopt manual prompts following Min et al. (2022) for exploration. For in-context learning (ICL), we randomly select 16 examples from the rest of the original sets. Besides, we keep the composition ratio of different labels as balanced as possible to weaken the influence of in-context learning (Tang et al., 2023).

Method	Accuracy	Decline	Method	Accuracy	Decline
LLaMA-7B	–	–	Alpaca-7B	51.30	32.47
w/ ICL	56.65	1.00	w/ ICL	49.60	40.13
T5-XXL	69.50	\	Flan-T5-XXL	72.60	54.80
w/ ICL	50.00	\	w/ ICL	75.33	49.33
GPT-3 *davinci*	–	–	ChatGPT	72.20	26.27
w/ ICL	63.00	\	w/ ICL	75.40	15.87

Table 2: Performance on HANS of different LLMs, – denotes this setting does not support our evaluation, \ denotes that no decline exists.

3.2. Results and Analysis

As mentioned in Section 2.1, the language models will exploit the overlap bias to make predictions, i.e., if the premise and hypothesis sentences contain many word overlaps, they will tend to predict the label as *entailment*. The performance on examples with label *non-entailment* will be worse than those with label *entailment*. As a result, we report the corresponding performance decline between label *non-entailment* and *entailment* to represent the extent of shortcut learning. Notice the more significant the decline is, the more seriously the language model suffers from shortcut learning.

Do recent LLMs have shortcut learning behaviors? We present the results in Table 2, we can find that: (1) LLMs without instruction tuning or RLHF can not directly support our evaluation in zero-shot setting, except T5[2]. There is no evident decline between different labels in the few-shot setting, but the overall accuracy is relatively low. (2) LLMs after instruction tuning or RLHF significantly aggravate the performance decline, and adopting in-context learning does not alleviate this problem effectively. (3) ChatGPT has the most potential to

[1] https://chat.openai.com/chat

[2] Thanks to the unified text-to-text pre-training paradigm, we can directly adopt T5 to predict the specific label with the input format following Raffel et al. (2020).

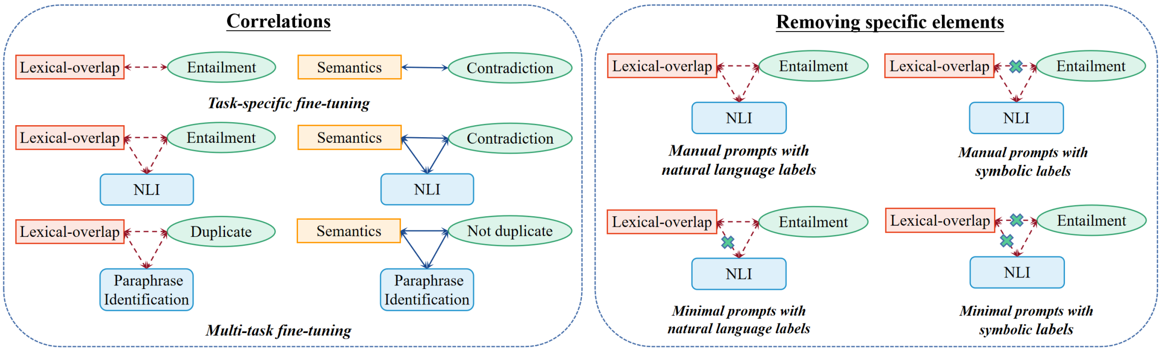

Figure 1: Left: correlations learned in different fine-tuning methods. Dashed line denotes spurious correlations, NLI denotes natural language inference task. Right: adopting different prompts and labels to remove specific elements in spurious correlations, e.g., minimal prompts contain no natural language instructions for specific tasks, symbolic labels are irrelevant to the previous ones adopted in specific tasks.

solve shortcut learning since the accuracy is the highest (75.40) and the decline is the lowest (15.87). Overall, our findings provide evidence that **recent LLMs with instruction tuning or RLHF still have shortcut learning behaviors**.

When do recent LLMs get shortcut learning behaviors? As mentioned above, we evaluate the performance of LLMs without and with instruction tuning or RLHF, since the performance decline between different target labels is only evident in the latter ones, **we would rather attribute the shortcut learning behaviors to the instruction tuning or RLHF processes.** Recently, Tang et al. point out that LLMs may learn shortcuts through the examples in the demonstration. However, as shown in Table 2, shortcut learning is serious in zero-shot settings, indicating that LLMs have got shortcut learning behaviors before in-context learning.

Why do recent LLMs get shortcut learning behaviors? We further analyze the potential reasons why shortcut learning behaviors occur during instruction tuning or RLHF processes. We draw inspiration from the previous works exploring shortcut learning based on the pre-training and task-specific fine-tuning paradigm. During the task-specific fine-tuning process, the models learn the spurious correlation between non-robust features and labels, as {Feature ↔ Label}. When it comes to LLMs with instruction tuning or RLHF processes, we consider that LLMs learn more complex correlations as {Task ↔ Feature ↔ Label} since it is a multi-task scenario. Figure 1 presents several correlations. We notice that not all the correlations are beneficial for prediction (e.g., correlations with dashed lines in Figure 1). Compared to understanding semantics, language models tend to use spurious correlations for prediction, known as shortcut learning. For example, in our experiments, the models will predict the corresponding label as *Entailment* through shortcuts from manual prompts and sentence inputs, which provide task information as natural language inference and spurious features such as lexical overlap, respectively. In-context learning may not benefit or even deepen the performance decline by providing helpful task information (Pan et al., 2023) while encouraging the models to adopt such spurious correlations. **Therefore, we attribute the reason to the spurious correlations LLMs learned in the instruction tuning or RLHF processes**.

4. Potential Solutions to Forgetting Spurious Correlations for LLMs

As mentioned in the previous section, LLMs can learn the correlations as {Task ↔ Feature ↔ Label}, and this may be converted to {Natural Language Inference ↔ Lexical-overlap ↔ Entailment} when applied to HANS dataset. We assume that each specific element may help LLMs recall the spurious correlation learned during training. Therefore, we encourage the LLMs to forget this spurious correlation by removing the specific elements during inference. More specifically, since the {Feature} element can not be removed since it is contained in the test examples, we adopt different strategies to remove the other two during inference as shown in the right part of Figure 1. Next, we give more detailed introduction and analysis of these strategies. Since ChatGPT presents the most potential, we conduct our experiments based on ChatGPT.

Remove the task element: minimal prompts with natural language labels. Motivated by previous works (Min et al., 2022; Pan et al., 2023), we adopt minimal prompts which remove any natural language instructions for the task rather than the previous widely-used manual prompts which provide the task information, thus weakening the

Methods	ICL w/ 4-shot		ICL w/ 8-shot		ICL w/ 16-shot		ICL w/ 32-shot		ICL w/ 64-shot	
	Accuracy	Decline	Accuracy	Decline	Accuracy	Decline	Accuracy	Decline	Accuracy	Decline
①	72.35	20.30	72.85	23.30	72.35	23.10	72.65	26.70	76.20	24.80
②	70.00	18.80	68.10	29.40	71.00	28.80	74.70	15.00	**77.00**	**2.40**
③	58.05	\	62.40	6.00	65.60	1.60	**69.90**	**0.60**	**76.05**	**0.50**
④	44.70	\	63.05	\	**69.10**	**0.20**	**74.50**	**1.00**	–	–

Table 3: Results on HANS of several potential solutions by removing specific elements in the learned correlations as shown in Figure 1. ①: manual prompts with labels *Yes* and *No*, ②: minimal prompts with labels *Yes* and *No*, ③: minimal prompts with labels *A4* and *B6*, ④: manual prompts with labels *A4* and *B6*. Some potential results are in bold. \ denotes that no decline exists.

{Task ↔ Feature} in spurious correlations. Instead, we force LLMs to conduct task learning (Pan et al., 2023) to learn valuable task information through in-context learning. As a result, we replace the prompts used in 3.2 with minimal prompts and conduct experiments on HANS. Since the number of examples in the demonstration plays a key role for LLMs to conduct task learning (Pan et al., 2023), we conduct experiments based on different numbers of examples in the demonstration. Results are shown in Table 3 (②) , we can find that: (1) in-context learning with 64 examples in the demonstration (w/ 64-shot) can effectively mitigate shortcut learning as well as improve the performance, (2) when the number of examples in the demonstration is relatively small (w/ 32/16/8-shot), the decline on examples with the label *non-entailment* still exists, and shows an upward trend with k decreasing. This verifies that the ability of task learning heavily relies on the number of examples in the demonstration when adopting minimal prompts. We assume that when the model can not achieve enough task information through in-context learning (e.g., fewer examples in the demonstration provide limited helpful information for specific tasks), it will still utilize the knowledge learned in the instruction tuning or RLHF processes. Besides, since we adopt the labels as *yes*/*no* in minimal prompts, LLMs will adopt the correlations {Task ↔ Label} to recognize the task information, (i.e., these labels are closely related to the natural language inference task after adopting different prompt templates in the multi-task fine-tuning process), then still presents the shortcut learning behaviors during inference.

Remove the task and label elements: minimal prompts with symbolic labels. To further remove the label information in prompts, we adopt symbols without semantics as label choices rather than natural language labels (e.g., *positive*, *yes*, *true*). More specifically, we randomly sample a combination of letters and numbers (e.g., *A4*, *7X*), and then randomly sample a mapping between symbolic labels and the original natural language labels (Wei et al., 2023a). Table 3 (③) presents the results, we can find that: (1) As the performance decline on different labels is small, in-context learning with examples adopting minimal prompts and symbolic labels in the demonstration can effectively mitigate shortcut learning, (2) the overall accuracy declines as the number of examples in the demonstration decreasing, indicating that LLMs can not achieve enough task information (i.e., this prompting template fully transforms the original natural language inference to a new task which is not contained in the training process, LLMs learn this new task through the examples in the demonstration).

Remove the label element: manual prompts with symbolic labels. We also remove only label information by adopting manual prompts but transforming the labels to symbols (i.e., the combination of letters and numbers). Due to the length limitation, we can maximum give 32 examples in the demonstration. We present the results in Table 3 (④) . Compared with minimal prompts, we can find that (1) in-context learning with 16/32 examples in the demonstration (w/ 16/32-shot) can perform better to mitigate the shortcut learning, (2) the performance is comparable with minimal prompts with 8 examples in the demonstration, and declines significantly with 4 examples in the demonstration. Overall, compared with the above two methods, manual prompts with symbolic labels seem to have the most potential. However, we also find that adopting manual prompts for each example may provide strong task information, as well as bringing extra length overhead (e.g., manual prompts contain the extra task description compared to minimal prompts, 64 examples are out of length limits in our experiments). As a result, although the labels of all examples in the demonstration are transformed into symbols, LLMs still generate natural language predictions. For example, we adopt the following prompt: *{Hypothesis} {Premise} Do these sentences show entailment? The answer is {A4/B6}*, LLMs still predict the label as *yes* and *No*.

Potential solution: finding the balance of task information and spurious correlations. In general, removing the elements of spurious correlations directly and urging the LLMs to achieve task information through in-context learning can mitigate shortcut learning. However, it needs numerous examples in the demonstration to provide task information to maintain comparable performance, and the performance declines with fewer examples. Since there is a maximum length constraint for the inputs of LLMs (nearly 2,048 tokens for most of the LLMs), more examples in the demonstration mean longer input for LLMs. It is necessary to find mitigation solutions with fewer examples (8 or even 4) in the demonstration. **Based on our experiments, achieving enough task information through in-context learning while forgetting spurious correlations is critical to mitigating shortcut learning.**

5. Enhanced Strategies for LLMs to Learn from In-context Information

In this section, we aim to make LLMs achieve enough task information through only a few examples in the demonstration while avoiding adopting the learned spurious correlations to make predictions. Firstly, we propose two simple yet effective methods to realize this purpose (§5.1). Then, we give details of our experimental settings to evaluate our methods (§5.2). Finally, we present the results and further analysis (§5.3).

5.1. Methodology

As mentioned above, we aim to provide enough task information while forgetting the spurious correlations via in-context learning for LLMs. Therefore, based on the minima prompts with symbolic labels, we aim to provide extra task information from two different respects, i.e., the manual prompts and natural language labels. Specifically, we can replace several examples with manual prompts, called mixed prompts, or replace several symbolic labels of the examples in the demonstration with natural language labels, called mixed labels. However, we should constrain the proportion and format of the replaced examples to find the balance between task information and spurious correlations.

Datasets	Train	Dev	Test	Eval.	Demon.
HANS	30,000	30,000	-	1,000	29,000
PAWS	11,988	677	-	300	80
SST-2	47,350	873	1,821	1,000	138

Table 4: Data Statistics. **Eval.** denotes examples used for evaluation, **Demon.** denotes candidate examples used for in-context learning.

Type	Prompt Template
Original minimal	Sentence 1: <Premise> Sentence 2: <Hypothesis> Label: *{A4/B6}*
Mixed labels	Sentence 1: <Premise> Sentence 2: <Hypothesis> Label: *{(Yes,True,A4,7X)/(No,False,B6,9Y)}*
Mixed prompts	Given following sentence 1 and sentence 2, if they are entailment, the answer is *A4*, if they are not entailment, the answer is *B6*. Sentence 1: <Premise> Sentence 2:<Hypothesis> Label: *{A4/B6}*

Table 5: Prompts format of our methods applied in in-context learning, Mixed Prompts only present the first one and others are original minimal prompts.

Mixed prompts. To avoid too much task information for LLMs to memorize the previous spurious correlations, we only replace one example with the manual prompt. Motivated by mixed prompts in previous works (Kojima et al., 2022; Zhang et al., 2023), we replace the first example in the demonstration and find this effective enough to provide task information. Specifically, take the natural language inference task as an example, given one test instance $I_i = (x_i, y_i)$, k examples in the demonstration, the model predicts the label formatted as:

$$P(y_i|x_i) = P(y_i|N(I_1) \oplus I_2, ..., \oplus I_k \oplus x_i), \qquad (2)$$

where $\oplus$ denotes the concatenation operation, $N(\cdot)$ denotes the manual prompt, and we omit the minimal prompt $M(\cdot)$ in other instances (e.g., $M(I_2)$).

Mixed labels. After adopting minimal prompts, we transform the original labels (e.g., *Entailment* and *Non-entailment*) to several label sets (e.g., {*Yes, True, A4, 7X*} and {*No, False, B6, 9Y*}), denoted as *Entailment* set and *Non-entailment* set, respectively. Then, we replace the original example labels in the demonstration with labels from the corresponding label set in a particular proportion. The part of natural language labels provides task information, while the composition of symbolic labels avoids LLMs using spurious correlations and learns from in-context information. We consider a prediction to be correct if the label predicted by the model is in the *Entailment* set and the ground-truth label is also *Entailment*, and vice versa. Notice the different compositions of label sets may have different effects, we will explore this more in Section 5.3.

5.2. Experimental settings

We evaluate the effectiveness of our methods on three well-studied tasks, i.e., natural language inference (NLI), sentiment analysis, and paraphrase identification. As shown in Figure 1, we study two well-known biases. For lexical-overlap bias, we adopt HANS (McCoy et al., 2020) and PAWS$_{\text{QQP}}$ (Zhang et al., 2019) datsets. HANS consists of pairs of premise and hypothesis sentences

Methods	ICL w/ 4-shot		ICL w/ 8-shot		ICL w/ 16-shot		ICL w/ 32-shot	
	Accuracy	Decline	Accuracy	Decline	Accuracy	Decline	Accuracy	Decline
HANS								
Manual Prompts	72.35	20.30	72.85	23.30	72.35	23.10	72.65	26.70
Minimal Propmts	52.83	\	59.10	2.20	62.25	2.70	68.90	2.00
Mixed Prompts	**73.73**	\	71.10	1.40	70.63	\	73.30	6.60
Mixed Labels	53.20	\	69.02	\	68.30	\	**74.20**	\
PAWS								
Manual Prompts	79.50	21.66	80.33	15.34	81.17	14.34	79.50	7.00
Minimal Propmts	61.67	\	66.00	\	76.00	\	78.33	0.67
Mixed Prompts	**85.33**	**6.00**	**85.17**	**2.34**	**83.50**	**1.67**	**83.33**	**3.34**
Mixed Labels	62.60	\	76.30	\	77.20	\	79.70	\
SST-2								
Manual Prompts	87.80	11.80	89.50	10.40	90.55	8.30	92.70	7.20
Minimal Propmts	49.90	25.40	88.55	8.50	95.70	2.40	96.45	1.30
Mixed Prompts	**95.65**	**2.10**	**96.13**	**0.85**	**96.85**	**1.90**	**96.60**	**1.60**
Mixed Labels	78.80	\	90.90	\	95.20	\	94.50	\

Table 6: Results of different prompts and our methods. Manual Prompts and Minimal Prompts denote two baselines as mentioned in the main body. Our methods are based on original minimal prompts with symbolic labels. Some significant results of our methods are in bold. \ denotes that no decline exists.

whose labels are *Entailment* and *Non-entailment*. PAWS$_{\mathrm{QQP}}$ is a set in which the question pairs are highly overlapping in words. Since the models tend to exploit lexical overlaps to predict the positive labels (e.g., *Entailment* and *Duplicate*), we compare the performance of different labels. For single-word bias, we use SST-2 (Socher et al., 2013) dataset, which is a binary sentiment classification task based on movie reviews. We first select the examples containing specific words (e.g., film, movie) following the previous works (Si et al., 2023). Then, we split these examples into two sets according to their labels. We also compare the performance of different labels to show if these specific words are related to specific labels.

We adopt ChatGPT for all experiments since ChatGPT achieves the highest accuracy and presents the most potential in alleviating the shortcut learning as shown in Table 2. Due to budget limitations, i.e., we can only conduct experiments through Openai Api since ChatGPT has not been opened publicly, we select part of the examples to serve as our test sets. We randomly select the examples with an equal proportion of different labels for all datasets. Specifically, we keep the same setting as mentioned in Section 3.1 for HANS. Since PAWS has 486 examples with label *Not duplicate* and 191 with label *Duplicate*, we randomly select 150 examples for each label and 300 examples in total to keep the number of examples of each label the same. For SST-2, we first select the specific words related to the labels, then we choose examples containing at least two such words from original test sets, including 1,138 examples in total. Then, we select 1000 examples in total and keep the numbers of examples for each label the same (i.e., the numbers for examples with label *Positive* and *Negative* are both 500). We randomly select examples from the rest of the set for in-context learning. Table 4 lists the detailed data statistics.

We conduct multiple experiments for our methods to avoid the influence of randomness. We adopt several manual and minimal prompts with different symbolic labels for our baseline settings and mixed prompts to conduct experiments and report the average performance. Besides, we adopt different label choices and compositions for the mixed labels strategy and report the average performance. Table 5 present particle prompts we adopted in our experiments, and more details can be found in the previous work (Min et al., 2022).

5.3. Results and Analysis

Main results. We adopt mixed prompts and mixed labels based on minimal prompts and compare with two baseline settings, i.e., adopting manual prompts with natural language labels and minimal prompts with symbolic labels, called manual baseline and minimal baseline in the following contexts, respectively. Table 6 presents all results. We report the overall accuracy and performance decline of two labels to measure the extent of shortcut learning and verify the effectiveness of our methods. We can find that: (1) **Our methods can effectively mitigate shortcut learning on all datasets.** The performance decline of two labels with our methods is much smaller than those in manual baselines, even with 8/16 examples in the demonstration. (2) **Our methods are all better than minimal baselines for overall performance.** Compared with manual baselines, our methods perform better in most testing cases on PAWS and SST-2, and better with 32 examples in the demonstration on HANS. Notice researchers may overestimate the performance of manual baselines since LLMs are easy to predict one specific label with spurious correlations. (3) **Mixed labels can achieve promising performance with a few examples, while mixed prompts can perform well with fewer examples. Mixed prompts are more effective than mixed labels.** As mentioned in Section 3, manual prompts can provide more information than natural language labels through in-context learning. Therefore, mixed prompts need relatively fewer examples to be effective than mixed labels.

Can mixed prompts perform better with constraints? Following Zhang et al. (2023), we further add task-specific constraints, e.g., for HANS dataset, based on the task information: *Given following sentence 1 and sentence 2, if they are entailment, the answer is A4, if they are not entailment, the answer is B6.*, we add the following constraint: *Consider the actual semantics and do not focus on the word overlaps.* We compare the results of original mixed prompts with constraints in Figure 2. **Adopting the mixed prompts method further with task-specific constraints does not bring many benefits in mitigating shortcut learning.** Besides, the overall performance even declines with few examples in the demonstration.

Do different label sets affect the performance? As mentioned in Section 5.1, we adopt a set of labels to transform the original labels, and the label sets contain both natural language and symbolic labels playing different roles during inference. We study the effects of the different compositions of the mixed labels. Firstly, we adopt all-natural language labels in the labels sets, denoted as natural language sets, e.g., {*Yes*, *True*, *Entailment*} and {*No*, *False*, *Non-entailment*}, and symbolic sets, e.g., {*A4*, *7X*, *F8*} and {*B6*, *9Y*, *D6*}. Figure 3 presents the results. Compared with our balanced sets, **adopting natural language sets deepens shortcut learning, and minimal sets lead to a decline in overall performance.** This further verifies the necessity of mixed labels and natural language and symbolic labels play different roles.

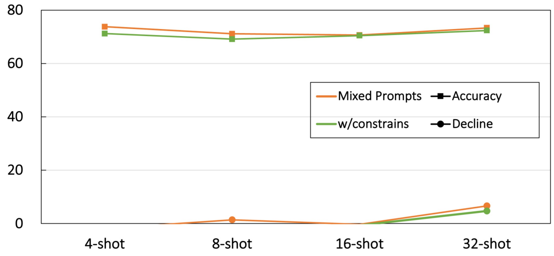

Figure 2: Results of mixed prompts with and without constraints.

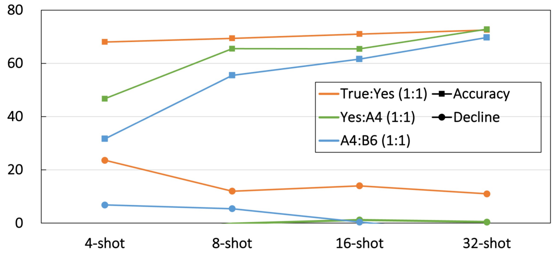

Figure 3: Results of different label sets.

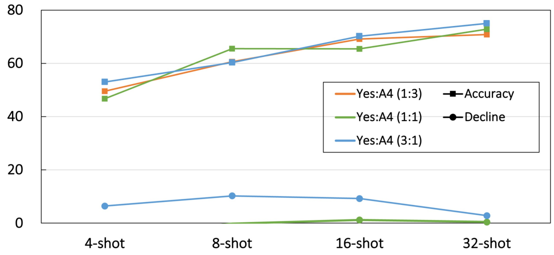

Figure 4: Results of different composition ratios.

Do different composition ratios affect the performance? We also study the effects of different composition ratios of the examples in the demonstration. We adopt the same proportion of various labels in our experiments mentioned in Section 5.3. We further consider two settings, we first fix the same label sets (e.g., {*Yes*, *A4*} and {*No*, *B6*}), then give different ratios to natural language labels. Results are shown in Figure 4 and we find that **adopting a high proportion of natural language labels leads to a more significant performance decline while adopting a low ratio can be helpful to the overall performance,** e.g., with 16 examples in the demonstration.

6. Conclusion

In this paper, we first verify that LLMs after instruction tuning or RLHF still suffer from shortcut learning from analytical experiments. Then, we further propose a framework for encouraging LLMs to **F**orget **S**purious correlations and **L**earn from **I**n-context information (FSLI) through two simple yet effective methods. Extensive experiments on three different tasks demonstrate that FSLI can effectively mitigate shortcut learning and improve overall performance.

Considering that shortcut learning can not be reflected in normal testing scenarios but truly hurts the generalization and performance in real-world settings, researchers should consider this problem and design more detailed and thorough evaluation methods. In the future, we will explore shortcut learning for more tasks, such as natural language generation and image classification.

7. Acknowledgements

We would like to thank the anonymous reviewers for their constructive comments. This work was supported by the National Science Foundation of China (NSFC No. 62206194) and the Natural Science Foundation of Jiangsu Province, China (Grant No. BK20220488).

8. Bibliographical References

Yejin Bang, Samuel Cahyawijaya, Nayeon Lee, Wenliang Dai, Dan Su, Bryan Wilie, Holy Lovenia, Ziwei Ji, Tiezheng Yu, Willy Chung, et al. 2023. A multitask, multilingual, multimodal evaluation of chatgpt on reasoning, hallucination, and interactivity. *arXiv preprint arXiv:2302.04023*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, et al. 2023. Sparks of artificial general intelligence: Early experiments with gpt-4. *arXiv preprint arXiv:2303.12712*.

Sebastien Bubeck and Mark Sellke. 2023. A universal law of robustness via isoperimetry. In *Advances in Neural Information Processing Systems*.

Jiangjie Chen, Wei Shi, Ziquan Fu, Sijie Cheng, Lei Li, and Yanghua Xiao. 2023. Say what you mean! large language models speak too positively about negative commonsense knowledge. *arXiv preprint arXiv:2305.05976*.

Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. 2017. Deep reinforcement learning from human preferences. *Advances in neural information processing systems*, 30.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2022. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Zhiyong Wu, Baobao Chang, Xu Sun, Jingjing Xu, and Zhifang Sui. 2022. A survey for in-context learning. *arXiv preprint arXiv:2301.00234*.

Mengnan Du, Fengxiang He, Na Zou, Dacheng Tao, and Xia Hu. 2022. Shortcut learning of large language models in natural language understanding: A survey. *arXiv preprint arXiv:2208.11857*.

Jacob Eisenstein. 2022. Uninformative input features and counterfactual invariance: Two perspectives on spurious correlations in natural language. *arXiv preprint arXiv:2204.04487*.

Dan Friedman, Alexander Wettig, and Danqi Chen. 2022. Finding dataset shortcuts with grammar induction. *arXiv preprint arXiv:2210.11560*.

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 *EMNLP*.

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.

Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel R Bowman, and Noah A Smith. 2018. Annotation artifacts in natural language inference data. In *NAACL HLT 2018*, pages 107–112.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. 2022. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*.

Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. 2019. Adversarial examples are not bugs, they are features. *Advances in neural information processing systems*, 32.

Nitish Joshi, Xiang Pan, and He He. 2022. Are all spurious features in natural language alike? an analysis through a causal lens. *arXiv preprint arXiv:2210.14011*.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. In *Advances in Neural Information Processing Systems*.

Yuxuan Lai, Chen Zhang, Yansong Feng, Quzhe Huang, and Dongyan Zhao. 2021. Why machine reading comprehension models learn shortcuts? *arXiv preprint arXiv:2106.01024*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *ACL*, pages 7871–7880.

Alisa Liu, Zhaofeng Wu, Julian Michael, Alane Suhr, Peter West, Alexander Koller, Swabha Swayamdipta, Noah A Smith, and Yejin Choi. 2023. We're afraid language models aren't modeling ambiguity. *arXiv preprint arXiv:2304.14399*.

R Thomas McCoy, Ellie Pavlick, and Tal Linzen. 2020. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In *57th Annual Meeting of the Association for Computational Linguistics, ACL 2019*, pages 3428–3448. Association for Computational Linguistics (ACL).

Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2022. Rethinking the role of demonstrations: What makes in-context learning work? *arXiv preprint arXiv:2202.12837*.

OpenAI. 2023. Gpt-4 technical report.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744.

Jane Pan, Tianyu Gao, Howard Chen, and Danqi Chen. 2023. What in-context learning" learns" in-context: Disentangling task recognition and task learning. *arXiv preprint arXiv:2305.09731*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551.

Alexis Ross, Matthew E Peters, and Ana Marasović. 2022. Does self-rationalization improve robustness to spurious correlations? *arXiv preprint arXiv:2210.13575*.

Victor Sanh, Albert Webson, Colin Raffel, Stephen H Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, et al. 2021. Multitask prompted training enables zero-shot task generalization. *arXiv preprint arXiv:2110.08207*.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.

Roy Schwartz and Gabriel Stanovsky. 2022. On the limitations of dataset balancing: The lost battle against spurious correlations. *arXiv preprint arXiv:2204.12708*.

Murray Shanahan. 2022. Talking about large language models. *arXiv preprint arXiv:2212.03551*.

Chenglei Si, Dan Friedman, Nitish Joshi, Shi Feng, Danqi Chen, and He He. 2023. What spurious features can pretrained language models combat?

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1631–1642.

Joe Stacey, Pasquale Minervini, Haim Dubossarsky, Sebastian Riedel, and Tim Rocktäschel. 2020. Avoiding the hypothesis-only bias in natural language inference via ensemble adversarial training. *arXiv preprint arXiv:2004.07790*.

Ruixiang Tang, Dehan Kong, Longtao Huang, and Hui Xue. 2023. Large language models can be lazy learners: Analyze shortcuts in in-context learning. *arXiv preprint arXiv:2305.17256*.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B Hashimoto. 2023. Alpaca: A strong, replicable instruction-following model. *Stanford Center for Research* *on Foundation Models. https://crfm. stanford. edu/2023/03/13/alpaca. html*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Lifu Tu, Garima Lalwani, Spandana Gella, and He He. 2020. An empirical study on robustness to spurious correlations using pre-trained language models. *Transactions of the Association for Computational Linguistics*, 8:621–633.

Prasetya Ajie Utama, Nafise Sadat Moosavi, and Iryna Gurevych. 2020. Towards debiasing nlu models from unknown biases. *arXiv preprint arXiv:2009.12303*.

Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2021. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, et al. 2022. Emergent abilities of large language models. *Transactions on Machine Learning Research*.

Jerry Wei, Le Hou, Andrew Lampinen, Xiangning Chen, Da Huang, Yi Tay, Xinyun Chen, Yifeng Lu, Denny Zhou, Tengyu Ma, et al. 2023a. Symbol tuning improves in-context learning in language models. *arXiv preprint arXiv:2305.08298*.

Jerry Wei, Jason Wei, Yi Tay, Dustin Tran, Albert Webson, Yifeng Lu, Xinyun Chen, Hanxiao Liu, Da Huang, Denny Zhou, et al. 2023b. Larger language models do in-context learning differently. *arXiv preprint arXiv:2303.03846*.

Jiaxin Wen, Yeshuang Zhu, Jinchao Zhang, Jie Zhou, and Minlie Huang. 2022. Autocad: Automatically generating counterfactuals for mitigating shortcut learning. *arXiv preprint arXiv:2211.16202*.

Huaxiu Yao, Yu Wang, Sai Li, Linjun Zhang, Weixin Liang, James Zou, and Chelsea Finn. 2022. Improving out-of-distribution robustness via selective augmentation. In *International Conference on Machine Learning*, pages 25407–25437. PMLR.

Aohan Zeng, Xiao Liu, Zhengxiao Du, Zihan Wang, Hanyu Lai, Ming Ding, Zhuoyi Yang, Yifan Xu, Wendi Zheng, Xiao Xia, et al. 2022. Glm-130b: An open bilingual pre-trained model. *arXiv preprint arXiv:2210.02414*.

Yuan Zhang, Jason Baldridge, and Luheng He. 2019. Paws: Paraphrase adversaries from word scrambling. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1298–1308.

Yuhui Zhang, Michihiro Yasunaga, Zhengping Zhou, Jeff Z HaoChen, James Zou, Percy Liang, and Serena Yeung. 2023. Beyond positive scaling: How negation impacts scaling trends of language models. *arXiv preprint arXiv:2305.17311*.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. 2023. A survey of large language models. *arXiv preprint arXiv:2303.18223*.