Causal Prompting: Debiasing Large Language Model Prompting Based on Front-Door Adjustment

Congzhi Zhang[*1] **Linhai Zhang**[*2] **Jialong Wu**[*1] **Yulan He**[2, 3] **Deyu Zhou**[†1]

[1]School of Computer Science and Engineering, Key Laboratory of Computer Network and Information Integration, Ministry of Education, Southeast University, China
[2]Department of Informatics, King's College London, UK
[3]The Alan Turing Institute, UK
{zhangcongzhi, jialongwu, d.zhou}@seu.edu.cn
{linhai.zhang, yulan.he}@kcl.ac.uk

Abstract

Despite the notable advancements of existing prompting methods, such as In-Context Learning and Chain-of-Thought for Large Language Models (LLMs), they still face challenges related to various biases. Traditional debiasing methods primarily focus on the model training stage, including approaches based on data augmentation and reweighting, yet they struggle with the complex biases inherent in LLMs. To address such limitations, the causal relationship behind the prompting methods is uncovered using a structural causal model, and a novel causal prompting method based on front-door adjustment is proposed to effectively mitigate LLMs biases. In specific, causal intervention is achieved by designing the prompts without accessing the parameters and logits of LLMs. The chain-of-thought generated by LLM is employed as the mediator variable and the causal effect between input prompts and output answers is calculated through front-door adjustment to mitigate model biases. Moreover, to accurately represent the chain-of-thoughts and estimate the causal effects, contrastive learning is used to fine-tune the encoder of chain-of-thought by aligning its space with that of the LLM. Experimental results show that the proposed causal prompting approach achieves excellent performance across seven natural language processing datasets on both open-source and closed-source LLMs.

1 Introduction

Large Language Models (LLMs) have shown remarkable emergent abilities, including In-Context Learning (ICL) (Brown et al. 2020; Peng et al. 2024; Yang et al. 2024) and Chain-of-Thought (CoT) prompting (Wei et al. 2022; Wang et al. 2022), which allow LLMs to perform natural language tasks based on only a few instances without weight updating. These prompting methods have achieved significant results across many traditional natural language processing tasks, including sentiment analysis, natural language inference, and machine reading comprehension (Kojima et al. 2022; Zhou et al. 2022; Liu et al. 2023).

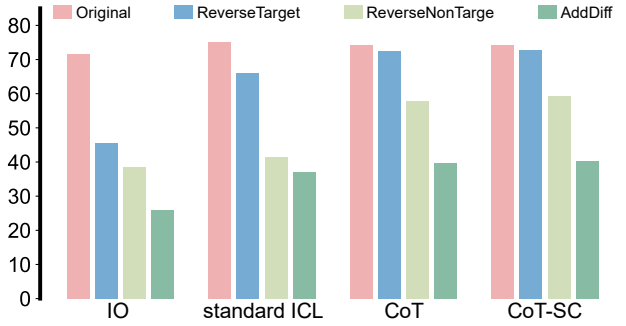

Figure 1: Performance of different prompting methods on ABSA (Pontiki et al. 2016) and its adversarial datasets on LLaMA-7b. ReverseTarget, ReverseNonTarget, and AddDiff denote three different adversarial transformations by TextFlint (Wang et al. 2021). IO denotes the zero-shot setting where only the input question outputs the answer.

However, recent studies have shown that these advanced prompting methods are not robust enough (Ye et al. 2023) and can lead LLMs to produce hallucinatory results with incorrect or unfaithful intermediate reasoning steps (Lyu et al. 2023; Wang et al. 2023b; Bao et al. 2024; Turpin et al. 2024; Wu et al. 2024d,e,c; Qin, Fang, and Xue 2024).

Some studies (Mallen et al. 2023; Wang et al. 2023d) believe that this phenomenon is due to the a conflict between the internal knowledge bias of LLMs and the external knowledge. Therefore, an effective solution is to interact with an external knowledge base to validate and adjust the reasoning process of LLMs (Wang et al. 2023b; Zhang et al. 2023). Moreover, recent work debiases the chain-of-thoughts of LLMs by incorporating counterfactual knowledge and causal interventions (Wu et al. 2024a). However, these methods are specifically tailored for knowledge-intensive tasks. Bias problems are also observed in other NLP tasks. As shown in Figure 2, in aspects-based sentiment analysis, mathematical reasoning, and multi-hop question-answering tasks, LLMs sometimes overly depend on certain text spans in the prompts, leading to wrong reasoning and answers. Notably, the first

[*]These authors contributed equally.
[†] Corresponding Author.

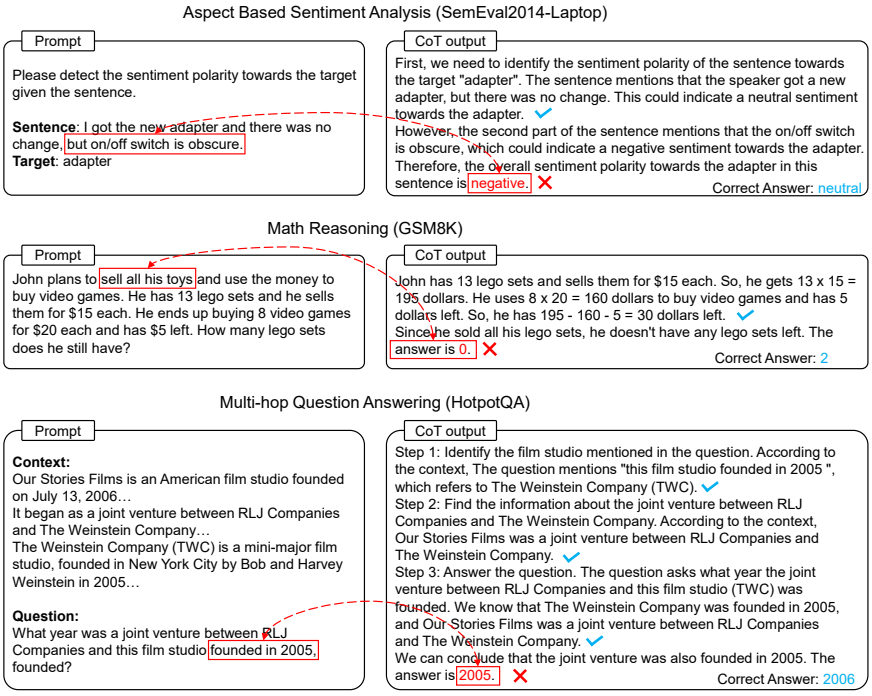

Figure 2: LLMs suffer from bias in the pertaining corpus, leading them to rely on irrelevant text spans in prompts and generating incoherent chain-of-thoughts that harm the logical reasoning capability of the model. These examples were obtained by using the CoT prompting (Wei et al. 2022) on the LLaMA3-8B model.

two tasks mentioned are not knowledge-intensive. We argue that LLMs fail to capture the true causal effect between questions and reasoning results and instead establish spurious correlations between certain text spans and answers.

In addition to the above qualitative analysis, our quantitative experiments also show that the current prompting methods are ineffective in addressing the bias issue. As shown in Figure 1, the performance of all prompting methods drops significantly when evaluated on the corresponding adversarial dataset compared to the original dataset, indicating that LLMs may suffer from bias in the pertaining corpus. Moreover, it has been demonstrated that LLMs exhibit label bias, recency bias, and entity bias from context (Zhao et al. 2021; Wang et al. 2023a; Fei et al. 2023).

Traditional debiasing methods mitigate the bias issue mainly during the model training stage, utilizing approaches such as data augmentation-based (Wei and Zou 2019; Lee et al. 2021) and reweighting (Schuster et al. 2019; Mahabadi, Belinkov, and Henderson 2019). Data augmentation-based methods face challenges due to the cost and complexity of annotating bias cases, particularly limited by context length. Reweight-based methods encounter difficulties in assigning weights to each sample in prompt-based learning scenarios. Recently, debias methods based on causal inference (Pearl et al. 2000; Pearl 2022) have become popular because of their strict theoretical guarantees and good generalization. Causal inference-based methods only need to calibrate model prediction results during the inference stage (Niu et al. 2021; Tian et al. 2022; Guo, Gong, and Lai 2022; Xu et al. 2023; Chen et al. 2023), which makes them well-suited for prompt-based learning scenarios. However, counterfactual inference requires accessing LLM output logits, while back-door adjustment requires specific confounding variable values.

To address the aforementioned challenge, we propose to debias prompting methods through causal intervention using front-door adjustment (Pearl, Glymour, and Jewell 2016). Front-door adjustment enables causal intervention without the need to access confounding variable values or LLM output logits. As shown in Figure 3(a), the causal relationship behind the prompting method is uncovered using a structural causal model. Here X denotes the input prompt, comprising demonstrations and test examples.

A denotes the predicted answer generated by the LLM. U is the unobservable confounder that introduces various biases in the pertaining corpus.

The debiasing process involves measuring the causal effect between the treatment X and the outcome A. However, as U absorbs complex biases of LLMs that are difficult to model or detect, back-door adjustment is not feasible for calculating the causal effect between X and A. To address this issue, as shown in Figure 3(b), we use the chain-of-thought generated by LLM as the mediator variable R between X and A.

As Figure 2 illustrates, while LLMs initially reason correctly, biases often confuse the final step of answer derivation. To simplify, we ignore the edges between U and R, aligning our causal graph with the front-door criterion (Pearl, Glymour, and Jewell 2016). By this way, we can use the front-door adjustment to estimate the causal effect between

X and A without accessing U.

Therefore, in this paper, we propose **Causal Prompting**, a novel prompting method for debiasing based on front-door adjustment. Unlike previous causal inference-based methods, causal intervention is implemented by modifying prompts without accessing the parameters and logits of LLMs. Specifically, to estimate the causal effect between X and R, we leverage self-consistency (SC) (Wang et al. 2022) of LLMs and a clustering algorithm to compute the probability of the chain-of-thought R. To measure the causal effect between R and A, we use the normalized weighted geometric mean (NWGM) approximation (Xu et al. 2015) to select the optimal demonstration set, which can help the model to generate an unbiased answer. Overall, CoT, SC, and ICL are effectively combined through front-door adjustment to mitigate LLM biases in NLP tasks. Note that in the clustering and NWGM algorithms, an $\mathrm{Encoder}$ is needed to obtain the representations of chain-of-thoughts. Since $\mathrm{Encoder}$ and LLMs have different semantic understanding of the chain-of-thought, we use contrastive learning (Chen et al. 2020) to fine-tune the $\mathrm{Encoder}$ to align its representation space with LLMs to estimate causal effects more accurately.

The contributions of this work are summarized as follows:

- Our work aims to identify and analyze the bias problem in LLM prompting methods from the perspective of causal inference, adhering more closely to the principles of the field. Moreover, the front-door adjustment is proposed to theoretically address the bias problem in prompting.
- Contrastive learning is proposed to fine-tune the $\mathrm{Encoder}$ of the chain-of-thoughts, aligning the space of the $\mathrm{Encoder}$ with LLMs to accurately capture representations of chain-of-thoughts and estimate causal effects.
- The proposed approach achieves excellent performance across seven natural language processing datasets using both open-source and closed-source LLMs.

2 Preliminaries

2.1 Structural Causal Model and Causal Intervention

A Structural Causal Model (SCM) (Pearl, Glymour, and Jewell 2016) is used to describe the causal relationships between variables. In SCM, we typically use a directed acyclic graph $G = \{V, E\}$, where V represents the set of variables and E represents the set of direct causal relationships.

As shown in Figure 3(a), X denotes the input prompt, including demonstrations and test examples. A denotes the predicted answer generated by the LLMs. LLMs generate answers based on prompt, so we have $X \rightarrow A$, which means that X is the direct cause of A. LLMs might learn spurious correlations between text patterns and answers from pre-trained corpora or instruction-supervised fine-tuning datasets (Xing et al. 2020; Li et al. 2024; Bao et al. 2024), leading to bias in downstream tasks. Previous work argues that the reason for this bias is that LLMs tend to follow a certain latent concept (Xie et al. 2021) or an implicit reasoning results (Li et al. 2024) in the reasoning process, rather than following the explicitly generated chain-of-thought. This

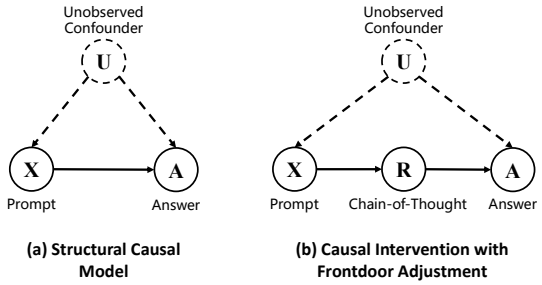

Figure 3: Structural causal model for the prompting method. (a) The causality of prompt and answer is confounded by unobservable variable. (b) The chain-of-thought generated by LLMs as a mediator variable between prompt and answer.

leads to the final answer does not necessarily follow from the generated chain-of-thought, specifically, there is no actual causal relationship between the chain-of-thought and the answer (Lyu et al. 2023; Bao et al. 2024). To accurately calculate the causal effect between X and A, we use the unobservable variable U to describe this latent concept or implicit reasoning results, using the back-door path $X \leftarrow U \rightarrow A$ denotes that the causality of X and A is confounded by U.

In SCM, if we want to compute the true causal effect between two variables X and A, we should block every back-door path between them (Pearl and Mackenzie 2018). For example, as shown in Figure 3(a), we should block $X \leftarrow U \rightarrow A$ to obtain the true causal effect between X and A. We typically use causal interventions for this purpose, which use the do operation to estimate the causal effect between X and A. In the causal graph satisfying Figure 3(a), the do-operation can be computed by back-door adjustment (Pearl, Glymour, and Jewell 2016):

$$P(A|do(X)) = \sum_{u} P(A|X, u)P(u) \tag{1}$$

2.2 Front-door Adjustment

Since confounding factor U is inaccessible, back-door adjustment cannot be performed. Fortunately, the front-door adjustment (Pearl, Glymour, and Jewell 2016) does not require access to the values of the confounding factor U to calculate the causal effect between X and A. As shown in Figure 3(b), we use the chain-of-thought generated by LLM as a mediator variable R between X and A.

In practice, as depicted in Figure 2, LLM can perform correct reasoning at the beginning, but it is often easily confused by bias in the last step of deriving the answer. Consequently, we decided to start with the simple SCM and focus on the confounder between X and A. In order to simplify the causal graph, we ignore the confounder of R with other variables, aligning our causal graph with the front-door criterion (Pearl, Glymour, and Jewell 2016). According to the front door adjustment, $P(A|do(X))$ can be formulated as:

$$P(A|do(X)) = \sum_{r} P(A|do(r))P(r|do(X)) \tag{2}$$

where $r \in R$ is the chain-of-thought generated by LLMs in response to the prompt X. The causal effect between

X and A is decomposed into two partially causal effects $P(r|do(X))$ and $P(A|do(r))$.

Next, we discuss how to estimate these two components separately. The first component is $P(r|do(X))$, represents the probability distribution of the chain-of-thought r given the intervention $do(X)$. To compute $P(r|do(X))$, we need to block the backdoor path $X \leftarrow U \rightarrow A \leftarrow R$ between X and R. Since there exists a collision structure $U \rightarrow A \leftarrow R$, the backdoor path has been blocked (Pearl, Glymour, and Jewell 2016) and we can get:

$$P(r|do(X)) = P(r|X) \tag{3}$$

Now, we focus on the computation of the second component $P(A|do(r))$, represents the probability distribution of the answer A given the intervention $do(r)$. To compute $P(A|do(r))$, we need to block the backdoor path $R \leftarrow X \leftarrow U \rightarrow A$ between R and A. Since we do not have access to the details of U, we implement back-door adjustments with the help of prompt X:

$$P(A|do(r)) = \sum_x P(x)P(A|r,x) \tag{4}$$

where $x \in X$ denotes the input prompt, including demonstrations and test examples.

Finally, substituting Equations (3) and (4) into Equation (2) after we obtain the estimation of $P(r|do(X))$ and $P(A|do(r))$. Hence, the final $P(A|do(X))$ can be represented as follows:

$$\begin{aligned} P(A|do(X)) &= \sum_r P(r|do(X))P(A|do(r)) \\ &= \underbrace{\sum_r P(r|X)}_{CoT-SC} \underbrace{\sum_x P(x)P(A|r,x)}_{ICL} \end{aligned} \tag{5}$$

where the first component $\sum_r P(r|do(X))$ can be estimated by combining the CoT and SC prompting methods, and the second component $P(A|do(r))$ can be computed by selecting the demonstration examples in ICL prompting.

3 Method

As shown in Figure 4, Causal Prompting aims to estimate the causal effect between input X and answer A. The estimation is achieved using the front-door adjustment, which divides the causal pathway into **two** distinct parts: the causal effect between X and chain-of-thought r, and the causal effect between r and A.

First, the causal effect between X and chain-of-thought r, $P(r|do(X))$ is estimated by combining the Chain-of-Thought prompting with a $\mathrm{Encoder}$-based clustering algorithm. **Second**, the causal effect between r and A, $P(A|do(r))$ is estimated by combining the In-Context Learning prompting with the normalized weighted geometric mean (NWGM) approximation algorithm. The final answer is aggregated by performing a weighted voting algorithm. Moreover, contrastive learning(Chen et al. 2020; Gao et al. 2022; Zhang, Zhang, and Zhou 2023) is employed to align the representation space of the $\mathrm{Encoder}$ and the LLMs for more precise estimation.

We will first introduce the estimation of $P(r|do(X))$ and $P(A|do(r))$, respectively, then combine them to derive $P(A|do(X))$. Finally, we will discuss how we align the representation space between the $\mathrm{Encoder}$ and the LLM.

3.1 Estimation of $P(r|do(X))$

We firstly undertake the estimation of $P(r|do(X))$. $P(r|do(X))$ measures the causal effect between input X and chain-of-thought r. As shown in Equation (3), the estimation of $P(r|do(X))$ is equivalent to the estimation of $P(r|X)$. However, $P(r|X)$ is still intractable for LLMs. On the one hand, the output probability is often inaccessible for most closed-source LLMs; on the other hand, the chain-of-thoughts r are challenging to enumerate comprehensively. Therefore, to estimate the causal effect $P(r|do(X))$ for both open-source and closed-source LLMs, we employ the CoT prompting and integrate it with a clustering algorithm. To be more specific, we initially prompt the LLMs to generate multiple CoTs based on the input. Subsequently, the CoTs are projected into embeddings. The embeddings are then clustered to form distinct groups based on their similarity. Finally, the centroid of each cluster is selected as the optimal and representative chain-of-thought. The probability associated with each representative chain-of-thought is then estimated based on the size of its respective cluster.

To enhance the quality of generated CoTs, n in-context demonstrations d are selected from training set based on question similarity. These demonstrations are then concatenated with the test question q^{test} to form the final prompt. Thus, the final prompt $\mathcal{P}$ is structured as follows:

$$\mathcal{P} = [d_1, ..., d_n, q^{test}] \tag{6}$$

where each $d_i = (q_i^{demo}, r_i^{demo})$ contain the demonstration question q_i^{demo} and its corresponding demonstration chain-of-thought r_i^{demo}. Where $i \in \{1, ..., n\}$, n denotes the number of demonstration examples in few-shot prompt method. In the practical implementation, we use prompt $\mathcal{P}$, which is fed into the LLMs to represent X in the structural causal model.

Based on the input prompt $\mathcal{P}$, LLMs are prompted to generate m distinct CoTs c by increasing the temperature parameter of LLMs. This adjustment encourages more diverse outputs, where the same procedure is also employed in self-consistency prompting of LLMs (Wang et al. 2023c). In this way, we can obtain the set of chain-of-thoughts as follows:

$$\{c_i | i = 1, ..., m\} = \mathrm{LLM}(\mathcal{P}) \tag{7}$$

To perform the distance-based clustering method, the generated CoT c_i are further fed into a $\mathrm{Encoder}$ to get the text embedding $\overline{c}_i$. Following the previous work (Devlin et al. 2018), the input is concatenated with the special tokens [CLS] and [SEP], and the embedding of the [CLS] token is taken as the embedding of CoT c_i.

$$\overline{c}_i = \mathrm{Encoder}([\text{CLS}], c_i, [\text{SEP}]) \tag{8}$$

Then K-means clustering algorithm (Har-Peled and Kushal 2005; Wu et al. 2023) is applied to the embeddings to get K clusters C as follows:

$$\{C_1, ..., C_K\} = \text{K-means}(\overline{c}_1, ..., \overline{c}_m) \tag{9}$$

where C_k refers to the k-th cluster of the clustering result, K denotes the number of clusters.

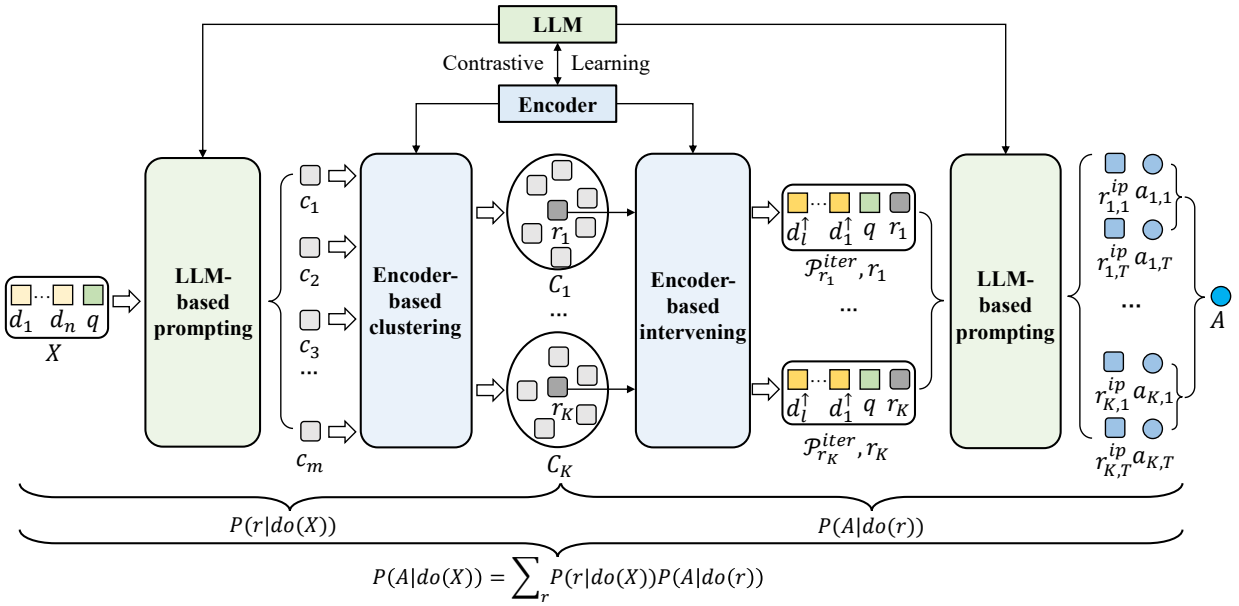

Figure 4: The overall framework of Causal Prompting. Firstly, based on the input prompt X consisting of the demonstration examples □ and a question □ of the test example, we query the LLM to generate m distinct CoTs □. Then, these CoTs are clustered into K clusters by an Encoder-based clustering algorithm. Subsequently, K representative CoTs ■ are selected by searching the closest CoT to the cluster center. Secondly, the optimal demonstration examples □ are retrieved for each representative CoT ■ through the Encoder-based intervention algorithm, and then the input prompt $\mathcal{P}_{r_k}^{iter}$ after the intervention is obtained. Finally, we query the LLM T times, obtaining T improved CoTs ■ and T answers ● for each representative CoT ■. The final answer ● is obtained by performing a weighted voting.

Based on the clusters, K representative chain-of-thoughts r are selected by searching the closest chain-of-thought to the cluster center.

$$r_k = \text{Center}(C_k), k = 1, ..., K \tag{10}$$

The causal effect between input X and chain-of-thought r_k is estimated based on the cluster size as follows:

$$P(r_k|do(X)) \approx \frac{|C_k|}{m} \tag{11}$$

where $|C_k|$ denotes the size of cluster C_k.

3.2 Estimation of $P(A|do(r))$

Based on the K chain-of-thoughts selected by Equation (10) in Section 3.1, we estimate $P(A|do(r_k))$ for each chain-of-thought r_k. For convenience, we omit the subscript k and use $P(A|do(r))$ to denote $P(A|do(r_k))$ in the following. $P(A|do(r))$ measures the causal effect between the chain-of-thought r and the answer A. Based on the discussion in Equation (4), $P(A|do(r))$ can be calculated with backdoor adjustment as follows:

$$P(A|do(r)) = \sum_{x \in X} P(x)P(A|r, x) = \mathbb{E}_{x \in X}[P(A|r, x)] \tag{12}$$

where $P(A|r, x)$ denotes the probability of the final answer A generated by LLM based on the given prompt x and the chain-of-thought r.

However, the value space of X is inexhaustible in most of the cases, and previous work employs the normalized weighted geometric mean (NWGM) approximation (Xu et al. 2015; Tian et al. 2022; Chen et al. 2023) to tackle this problem, where a confounder embedding $\overline{x}^{'}$ is estimated to approximate the expectation of variable X.

$$\mathbb{E}_{x \in X}[P(A|r, x)] \approx P(A|r, \mathbb{E}_{x \in X}[x]) \approx P(A|concat(r, \overline{x}^{'})) \tag{13}$$

where $concat(\cdot, \cdot)$ denotes vector concatenation, $\overline{x}^{'}$ denotes the confounder embedding of X.

Inspired by the previous works (Xu et al. 2015; Tian et al. 2022; Chen et al. 2023; Zhang, Zhang, and Zhou 2024), we propose a prompting version of NWGM approximation to perform the backdoor adjustment for LLMs prompting by combining a Encoder-based intervention and In-Context Learning (ICL) prompting. The original idea of NWGM is to augment the representation of the chain-of-thought r with an embedding $\overline{x}^{'}$ that contains all sample information as much as possible. However, at the prompting level, we cannot include all samples in context due to the limited context length, so we use only those samples that are most useful for improving the current chain-of-thought r.

Specifically, we use the Encoder to obtain the embedding $\overline{r}_k$ of the k-th chain-of-thought r_k. Subsequently, ICL demonstrations are selected by searching the entire training set based on the chain-of-thought embedding $\overline{r}_k$ to approximate the effect of taking expectations on input X. Finally, we rank the ICL demonstrations according to their similarity weights to indicate the importance of different samples.

Note that, as shown in Equation (6), the input prompt $\mathcal{P}$ includes demonstrations d and test question q^{test}. Directly modifying the certain text span of test examples will change the semantics of question q^{test}. Therefore, we only modify the demonstrations d and implement the NWGM approximation by In-Context Learning. In fact, the goal of our prompting version of the NWGM algorithm

is to enable the LLMs to learn from the demonstrations how to improve the chain-of-thought r of the test example. We introduce both wrong and correct chain-of-thoughts of demonstrations.

Given a training set $\mathcal{D} = \{d_j = (q_j, r_j^{wrong}, r_j^{correct})\}_{j=1}^{N}$, and a chain-of-thought r_k of test example, where q_j denotes the question of j-th training sample, r_j^{wrong} and $r_j^{correct}$ denote the wrong and correct chain-of-thoughts of demonstration d_j, N denotes the size of the training set, r_k refers to the k-th chain-of-thought selected by Equation (10) in Section 3.1. The embedding $\overline{r}_k$ of chain-of-thought r_k and the embedding $\overline{d}_j$ of demonstration d_j are obtained by the following:

$$\begin{aligned} \overline{r}_k &= \text{Encoder}([\text{CLS}], r_k, [\text{SEP}]) \\ \overline{d}_j &= \text{Encoder}([\text{CLS}], r_j^{wrong}, [\text{SEP}]) \end{aligned} \tag{14}$$

Previous works (Margatina et al. 2023; Liu et al. 2022) have shown that using demonstration examples that are semantically similar to the test examples allows better performance for In-Context Learning. Therefore, the back-door intervention is approximated by searching the most similar instance based on chain-of-thought embedding $\overline{r}_k$. Specifically, we sort the training set $\mathcal{D}$ from largest to smallest according to the cosine similarity between $\overline{r}_k$and $\overline{d}_j$.

$$\{d_j^{\uparrow}\}_{j=1}^{N} = Sort(\mathcal{D}, \overline{r}_k, \{\overline{d}_j\}_{j=1}^{N}) \tag{15}$$

where $d_j^{\uparrow}$ denotes the sorted demonstration example, $Sort$ means that, given a predefined cosine similarity function cos, the samples are ordered so that $cos(\overline{r}_k, \overline{d}_i) \geq cos(\overline{r}_k, \overline{d}_j)$ when $i < j$.

Then the l most similar demonstration examples are selected to concatenate into prompt, where $l \ll N$. Note that, unlike the KATE (Liu et al. 2021) method, we put the most similar demonstration samples closer to the test samples because this order is more beneficial for our NWGM algorithm to learn information for improving the chain-of-thoughts from the demonstration based on practical experiments. For each chain-of-thought r_k of a test sample, the final input prompt after intervention is given as follows:

$$\mathcal{P}_{r_k}^{iter} = [d_l^{\uparrow}, ..., d_1^{\uparrow}, q^{test}] \tag{16}$$

Subsequently, we query the LLMs T times, obtaining T answers and T improved chain-of-thoughts using the prompt $\mathcal{P}_{r_k}^{iter}$ and chain-of-thought r_k.

$$\{(r_{k,t}^{ip}, a_{k,t}) | t = 1, ..., T\} = \text{LLM}(\mathcal{P}_{r_k}^{iter}, r_k) \tag{17}$$

where $r_{k,t}^{ip}$ denotes the t-th improved chain-of-thought for chain-of-thought r_k.

We then use majority voting to estimate the probability of the answer as follows:

$$P(A|do(r_k)) \approx \frac{\sum_{t=1}^{T} \mathbb{I}(A = a_{k,t})}{T} \tag{18}$$

3.3 Estimation of $P(A|do(X))$

Based on the results of Equation (11) in Section 3.1 and Equation (18) in Section 3.2, the final answer is obtained by performing a weighted voting as follows:

$$\begin{aligned} P(A|do(X)) &= \sum_{r_k} P(r_k|do(X)) P(A|do(r_k)) \\ &= \sum_{k=1}^{K} \frac{|C_k|}{m} \cdot \frac{\sum_{t=1}^{T} \mathbb{I}(A = a_{k,t})}{T} \end{aligned} \tag{19}$$

Finally, we chose the answer with the largest weight as the final answer. In this way, with the front-door adjustment, we calibrate the probability distribution $P(A|X)$ obtained by the CoT-SC method to $P(A|do(X))$ obtained by the Causal Prompting method.

3.4 Representation Space Alignment

In the clustering discussed in Section 3.1 and NWGM algorithm presented in Section 3.2, an Encoder is needed to derive the representations of chain-of-thoughts. However, the semantic representation of Encoder and LLM differ significantly. Two chain-of-thoughts that LLM considers similar may not be close in the representation space of the Encoder. Through experiments we found that the chain-of-thoughts generated by LLM are not distinctly separable in the representation space of the vanilla Encoder.

To align the representation spaces of the Encoder and the LLMs, we take each chain-of-thought r in the training dataset $\mathcal{D}$ as an anchor, use LLM to generate the corresponding positive samples, use the other samples within the batch as negative samples, and then use contrastive learning to fine-tune the Encoder.

For chain-of-thought r, we prompt the LLM to generate a similar sentence r^+ as the positive sample. Following previous works (Gao et al. 2022; Zhang, Zhang, and Zhou 2023), we use the InfoNCE loss (Chen et al. 2020) to fine-tune the Encoder :

$$\sum_{\overline{r}_p \in Pos(r)} -log \frac{g(\overline{r}, \overline{r}_p)}{g(\overline{r}, \overline{r}_p) + \sum_{j \in Neg(r)} g(\overline{r}, \overline{r}_j)} \tag{20}$$

where the $\overline{r}$ and $\overline{r}_p$ are the representations of r and its positive samples. $Pos(r)$ and $Neg(r)$ refer to the positive set and the negative set for the chain-of-thought r. $Pos(r) = \{\overline{r}_{p1}, \overline{r}_{p2}\}$, where $\overline{r}_{p1}$ is augmented representation of the same chain-of-thought r, obtained with different dropout masks, and $\overline{r}_{p2}$ is the representation of positive sample r^+. $j \in Neg(r)$ is the index of in-batch negative samples. g is a function: $g(\overline{r}, \overline{r}_p) = exp(\overline{r}^T \overline{r}_p / temp)$, where $temp$ is a positive value of temperature in the contrastive learning.

4 Experiments

4.1 Datasets

We evaluate the effectiveness of our approach on three tasks: **Math Reasoning** (GSM8K (Cobbe et al. 2021), MATH (Hendrycks et al. 2021)), **Multi-hop Question Answering** (HotpotQA (Yang et al. 2018), MuSiQue (Trivedi et al. 2022)), and **Natural Language Understanding** (Aspect-based Sentiment Analysis (ABSA) (Pontiki et al. 2016), Natural Language Inference (NLI) (Williams, Nangia, and Bowman 2017), and Fact Verification (FV) (Thorne et al. 2018)). For the NLU tasks, we use the original datasets (in-distribution, ID) and the corresponding adversarial datasets (out-of-distribution, OOD) (Wang et al. 2021) to verify the robustness of our method.

4.2 Baselines

We compare our approach with three other few-shot prompting approaches to evaluate its effectiveness: Standard ICL (Brown et al. 2020), CoT (Wei et al. 2022) and CoT-SC (Wang et al. 2022).

4.3 Main Results

Table 1 shows the comparison results between causal prompting and the aforementioned baselines. Expectedly, the performance of Standard ICL, CoT, and CoT-SC improves progressively, as each subsequent method is an enhanced version of its predecessor. It not only confirms the effectiveness of integrating CoT into ICL, consistent with (Brown et al. 2020; Wei et al. 2022; Zhou et al. 2022), but also validates the efficacy of employing multiple sampling and voting strategies (Wang et al. 2022). **Causal Prompting** consistently delivers the best results across all metrics and datasets. It indicates that our prompting method can comprehensively improve the ability of LLM in all three tasks. Specifically, our method exhibits a more pronounced improvement in Math Reasoning and Multi-hop Question Answering tasks, with an average performance enhancement of

	GSM8K	MATH	HotpotQA		MuSiQue		ABSA	NLI	FV
Method	Acc	Acc	EM	F1	EM	F1	Acc	Acc	Acc
LLaMA2									
Standard ICL	6.14	3.71	41.20	59.56	26.09	41.16	47.26	28.20	56.87
CoT	27.07	4.72	44.70	64.84	18.71	30.27	49.12	27.56	70.07
CoT-SC	31.92	6.32	49.30	68.53	31.16	46.36	53.70	33.57	72.20
Causal Prompting	**36.47**	**8.76**	**52.20**	**70.88**	**34.68**	**48.79**	**67.55**	**50.83**	**81.07**
LLaMA3									
Standard ICL	18.65	14.24	37.20	62.17	17.42	24.22	72.14	63.75	80.67
CoT	74.07	40.35	48.90	72.75	38.88	54.38	71.55	64.19	81.80
CoT-SC	82.41	56.61	52.70	75.43	41.37	59.78	75.92	65.15	83.87
Causal Prompting	**87.95**	**62.76**	**58.50**	**78.18**	**48.07**	**64.23**	**79.06**	**67.97**	**86.67**
GPT-3.5									
Standard ICL	33.74	23.08	2.10	3.68	28.84	39.27	69.26	53.52	75.33
CoT	71.87	53.50	11.70	16.49	41.37	57.82	65.74	63.55	80.67
CoT-SC	80.21	58.38	41.60	56.82	46.27	60.83	74.59	66.88	82.73
Causal Prompting	**85.44**	**70.18**	**58.20**	**78.10**	**50.13**	**65.40**	**80.13**	**71.93**	**86.53**

Table 1: The comparison results of Causal Prompting against baselines across different backbone LLMs, including LLaMA2, LLaMA3 and GPT-3.5, on seven datasets. The best results are in bold.

	ABSA		NLI		FV	
Methods	Ori	Adv	Ori	Adv	Ori	Adv
ICL	75.71	70.30	76.30	50.27	90.00	76.00
CoT	77.27	68.60	74.81	54.77	91.40	77.00
CoT-SC	**80.56**	73.53	76.17	57.16	93.40	79.10
Ours	79.78	**78.69**	**76.67**	**58.62**	**95.40**	**82.30**

Table 2: The results of the robustness study on LLaMA3. Ori denotes the original dataset (ID) and Adv denotes the adversarial dataset (OOD). The best results are in bold.

approximately **5%-10%**. This substantial increase underscores our method's greater efficacy in tackling more challenging problems.

4.4 Robustness Study

Recent causal-based works (Tian et al. 2022; Zhang, Zhang, and Zhou 2024; Zhu et al. 2023; Wang et al. 2023a; Xu et al. 2023; Niu et al. 2021; Schuster et al. 2019; Wu et al. 2024b) have shown that using symmetric and adversarial (out-of-distribution) datasets **can evaluate the debiasing ability of models**. Following their practice, we evaluate **Causal Prompting** on both original data and adversarial data of the NLU tasks, respectively. Tables 2 show the performance comparison results of our method and baselines on LLaMA3 model. Although the performance of Causal Prompting decreases on Ori of ABSA, the improvement is larger on Adv data, resulting in the highest overall performance, see in Table 1. This phenomenon aligns with findings reported in previous work on causal inference (Tian et al. 2022; Wang et al. 2023a). It can be observed that the Adv of Causal Prompting is the highest on all datasets. This shows that our method generalizes well for both synthetic adversarial data in ABSA and NLI generated by TextFlint (Wang et al. 2021) and human-annotated real adversarial data in FV. This further validates the robustness of our model in handling datasets with significant bias.

5 Conclusion

We introduced Causal Prompting, a novel method for debiasing LLMs in NLP tasks by utilizing front-door adjustment in this work. The CoT generated by LLMs is employed as a mediator variable in the causal graph. Specifically, the causal effect between input prompt and output answer is decomposed into two distinct components, the causal effect from the input prompt to CoTs and from CoTs to the answer. The former component is estimated by combining the CoT prompting with a Encoder-based clustering algorithm. The latter component is estimated by combining the ICL prompting with the NWGM approximation algorithm. Moreover, Contrastive learning is used to fine-tune the Encoder so that the representation space of the Encoder is aligned with the LLM to estimate the causal effect more accurately. Our experimental results demonstrate that Causal Prompting significantly improves performance across seven NLP tasks on both open-source and closed-source LLMs. This approach, which both enhances performance and yields debiased responses, aligns with the trend of obtaining optimal results at test time. It can be extended to a broader range of scenarios, such as safety or alignment, under theoretical guidance.

Acknowledgments

The authors would like to thank the anonymous reviewers for their insightful comments. This work is funded by the National Natural Science Foundation of China (62176053). This work was supported in part by the UK Engineering and Physical Sciences Research Council (EPSRC) through a Turing AI Fellowship (grant no. EP/V020579/1, EP/V020579/2) and Innovate UK through the Accelerating Trustworthy AI programme (grant no. 10093055). This work is supported by the Big Data Computing Center of Southeast University.

References

Bao, G.; Zhang, H.; Yang, L.; Wang, C.; and Zhang, Y. 2024. LLMs with Chain-of-Thought Are Non-Causal Reasoners. *arXiv preprint arXiv:2402.16048*.

Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J. D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33: 1877–1901.

Chen, T.; Kornblith, S.; Norouzi, M.; and Hinton, G. 2020. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, 1597–1607. PMLR.

Chen, Z.; Hu, L.; Li, W.; Shao, Y.; and Nie, L. 2023. Causal intervention and counterfactual reasoning for multi-modal fake news detection. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 627–638.

Cobbe, K.; Kosaraju, V.; Bavarian, M.; Chen, M.; Jun, H.; Kaiser, L.; Plappert, M.; Tworek, J.; Hilton, J.; Nakano, R.; et al. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Fei, Y.; Hou, Y.; Chen, Z.; and Bosselut, A. 2023. Mitigating Label Biases for In-context Learning. *arXiv preprint arXiv:2305.19148*.

Gao, J.; Wang, W.; Yu, C.; Zhao, H.; Ng, W.; and Xu, R. 2022. Improving event representation via simultaneous weakly supervised contrastive learning and clustering. *arXiv preprint arXiv:2203.07633*.

Guo, W.; Gong, Q.; and Lai, H. 2022. Counterfactual Multihop QA: A Cause-Effect Approach for Reducing Disconnected Reasoning. *arXiv preprint arXiv:2210.07138*.

Har-Peled, S.; and Kushal, A. 2005. Smaller coresets for k-median and k-means clustering. In *Proceedings of the twenty-first annual symposium on Computational geometry*, 126–134.

Hendrycks, D.; Burns, C.; Kadavath, S.; Arora, A.; Basart, S.; Tang, E.; Song, D.; and Steinhardt, J. 2021. Measuring Mathematical Problem Solving With the MATH Dataset. *NeurIPS*.

Kojima, T.; Gu, S. S.; Reid, M.; Matsuo, Y.; and Iwasawa, Y. 2022. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35: 22199–22213.

Lee, M.; Won, S.; Kim, J.; Lee, H.; Park, C.; and Jung, K. 2021. CrossAug: A Contrastive Data Augmentation Method for Debiasing Fact Verification Models. In *Proceedings of the 30th ACM International Conference on Information amp; Knowledge Management*.

Li, Z.; Jiang, G.; Xie, H.; Song, L.; Lian, D.; and Wei, Y. 2024. Understanding and Patching Compositional Reasoning in LLMs. *arXiv preprint arXiv:2402.14328*.

Liu, J.; Shen, D.; Zhang, Y.; Dolan, B.; Carin, L.; and Chen, W. 2021. What Makes Good In-Context Examples for GPT-3? *arXiv preprint arXiv:2101.06804*.

Liu, J.; Shen, D.; Zhang, Y.; Dolan, B.; Carin, L.; and Chen, W. 2022. What Makes Good In-Context Examples for GPT-3? In Agirre, E.; Apidianaki, M.; and Vulić, I., eds., *Proceedings of Deep Learning Inside Out (DeeLIO 2022): The 3rd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*, 100–114. Dublin, Ireland and Online: Association for Computational Linguistics.

Liu, P.; Yuan, W.; Fu, J.; Jiang, Z.; Hayashi, H.; and Neubig, G. 2023. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*, 55(9): 1–35.

Lyu, Q.; Havaldar, S.; Stein, A.; Zhang, L.; Rao, D.; Wong, E.; Apidianaki, M.; and Callison-Burch, C. 2023. Faithful Chain-of-Thought Reasoning. In Park, J. C.; Arase, Y.; Hu, B.; Lu, W.; Wijaya, D.; Purwarianti, A.; and Krisnadhi, A. A., eds., *Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, 305–329. Nusa Dua, Bali: Association for Computational Linguistics.

Mahabadi, R.; Belinkov, Y.; and Henderson, J. 2019. End-to-End Bias Mitigation by Modelling Biases in Corpora. *arXiv: Computation and Language,arXiv: Computation and Language*.

Mallen, A.; Asai, A.; Zhong, V.; Das, R.; Khashabi, D.; and Hajishirzi, H. 2023. When Not to Trust Language Models: Investigating Effectiveness of Parametric and Non-Parametric Memories. In Rogers, A.; Boyd-Graber, J.; and Okazaki, N., eds., *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 9802–9822. Toronto, Canada: Association for Computational Linguistics.

Margatina, K.; Schick, T.; Aletras, N.; and Dwivedi-Yu, J. 2023. Active Learning Principles for In-Context Learning with Large Language Models. *arXiv preprint arXiv:2305.14264*.

Niu, Y.; Tang, K.; Zhang, H.; Lu, Z.; Hua, X.-S.; and Wen, J.-R. 2021. Counterfactual vqa: A cause-effect look at language bias. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 12700–12710.

Pearl, J. 2022. Direct and indirect effects. In *Probabilistic and causal inference: the works of Judea Pearl*, 373–392.

Pearl, J.; Glymour, M.; and Jewell, N. P. 2016. *Causal inference in statistics: A primer*. John Wiley & Sons.

Pearl, J.; and Mackenzie, D. 2018. *The book of why: the new science of cause and effect*. Basic books.

Pearl, J.; et al. 2000. Models, reasoning and inference. *Cambridge, UK: CambridgeUniversityPress*, 19(2): 3.

Peng, Y.; Hu, X.; Peng, J.; Geng, X.; Yang, X.; et al. 2024. LIVE: Learnable In-Context Vector for Visual Question Answering. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Pontiki, M.; Galanis, D.; Papageorgiou, H.; Androutsopoulos, I.; Manandhar, S.; AL-Smadi, M.; Al-Ayyoub, M.; Zhao, Y.; Qin, B.; De Clercq, O.; et al. 2016. Semeval-2016 task 5: Aspect based sentiment analysis. In *ProWorkshop on Semantic Evaluation (SemEval-2016)*, 19–30. Association for Computational Linguistics.

Qin, Y.; Fang, P.; and Xue, H. 2024. PEARL: Input-Agnostic Prompt Enhancement with Negative Feedback Regulation for Class-Incremental Learning. arXiv:2412.10900.

Schuster, T.; Shah, D. J.; Yeo, Y. J. S.; Filizzola, D.; Santus, E.; and Barzilay, R. 2019. Towards debiasing fact verification models. *arXiv preprint arXiv:1908.05267*.

Thorne, J.; Vlachos, A.; Christodoulopoulos, C.; and Mittal, A. 2018. FEVER: a large-scale dataset for fact extraction and VERification. *arXiv preprint arXiv:1803.05355*.

Tian, B.; Cao, Y.; Zhang, Y.; and Xing, C. 2022. Debiasing NLU Models via Causal Intervention and Counterfactual Reasoning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 11376–11384.

Trivedi, H.; Balasubramanian, N.; Khot, T.; and Sabharwal, A. 2022. MuSiQue: Multihop Questions via Single-hop Question Composition. *Transactions of the Association for Computational Linguistics*.

Turpin, M.; Michael, J.; Perez, E.; and Bowman, S. 2024. Language models don't always say what they think: unfaithful explanations in chain-of-thought prompting. *Advances in Neural Information Processing Systems*, 36.

Wang, F.; Mo, W.; Wang, Y.; Zhou, W.; and Chen, M. 2023a. A Causal View of Entity Bias in (Large) Language Models. In Bouamor, H.; Pino, J.; and Bali, K., eds., *Findings of the Association for Computational Linguistics: EMNLP 2023*, 15173–15184. Singapore: Association for Computational Linguistics.

Wang, K.; Duan, F.; Wang, S.; Li, P.; Xian, Y.; Yin, C.; Rong, W.; and Xiong, Z. 2023b. Knowledge-driven cot: Exploring faithful reasoning in llms for knowledge-intensive question answering. *arXiv preprint arXiv:2308.13259*.

Wang, X.; Liu, Q.; Gui, T.; Zhang, Q.; Zou, Y.; Zhou, X.; Ye, J.; Zhang, Y.; Zheng, R.; Pang, Z.; et al. 2021. Textflint: Unified multilingual robustness evaluation toolkit for natural language processing. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations*, 347–355.

Wang, X.; Wei, J.; Schuurmans, D.; Le, Q.; Chi, E.; Narang, S.; Chowdhery, A.; and Zhou, D. 2022. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*.

Wang, X.; Wei, J.; Schuurmans, D.; Le, Q. V.; Chi, E. H.; Narang, S.; Chowdhery, A.; and Zhou, D. 2023c. Self-Consistency Improves Chain of Thought Reasoning in Language Models. In *The Eleventh International Conference on Learning Representations*.

Wang, Y.; Feng, S.; Wang, H.; Shi, W.; Balachandran, V.; He, T.; and Tsvetkov, Y. 2023d. Resolving knowledge conflicts in large language models. *arXiv preprint arXiv:2310.00935*.

Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Xia, F.; Chi, E.; Le, Q. V.; Zhou, D.; et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35: 24824–24837.

Wei, J.; and Zou, K. 2019. EDA: Easy Data Augmentation Techniques for Boosting Performance on Text Classification Tasks. 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)*.

Williams, A.; Nangia, N.; and Bowman, S. R. 2017. A broad-coverage challenge corpus for sentence understanding through inference. *arXiv preprint arXiv:1704.05426*.

Wu, J.; Yu, T.; Chen, X.; Wang, H.; Rossi, R.; Kim, S.; Rao, A.; and McAuley, J. 2024a. DeCoT: Debiasing Chain-of-Thought for Knowledge-Intensive Tasks in Large Language Models via Causal Intervention. In Ku, L.-W.; Martins, A.; and Srikumar, V., eds., *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 14073–14087. Bangkok, Thailand: Association for Computational Linguistics.

Wu, J.; Zhang, L.; Zhou, D.; and Xu, G. 2024b. DINER: Debiasing Aspect-based Sentiment Analysis with Multi-variable Causal Inference. *arXiv preprint arXiv:2403.01166*.

Wu, S.; Lu, K.; Xu, B.; Lin, J.; Su, Q.; and Zhou, C. 2023. Self-Evolved Diverse Data Sampling for Efficient Instruction Tuning. *arXiv preprint arXiv:2311.08182*.

Wu, Y.; Hu, X.; Sun, Y.; Zhou, Y.; Zhu, W.; Rao, F.; Schiele, B.; and Yang, X. 2024c. Number it: Temporal Grounding Videos like Flipping Manga. *arXiv preprint arXiv:2411.10332*.

Wu, Y.; Zhou, S.; Yang, M.; Wang, L.; Zhu, W.; Chang, H.; Zhou, X.; and Yang, X. 2024d. Unlearning Concepts in Diffusion Model via Concept Domain Correction and Concept Preserving Gradient. *arXiv preprint arXiv:2405.15304*.

Wu, Y.; Zhu, W.; Cao, J.; Lu, Y.; Li, B.; Chi, W.; Qiu, Z.; Su, L.; Zheng, H.; Wu, J.; et al. 2024e. Video Repurposing from User Generated Content: A Large-scale Dataset and Benchmark. *arXiv preprint arXiv:2412.08879*.

Xie, S. M.; Raghunathan, A.; Liang, P.; and Ma, T. 2021. An explanation of in-context learning as implicit bayesian inference. *arXiv preprint arXiv:2111.02080*.

Xing, X.; Jin, Z.; Jin, D.; Wang, B.; Zhang, Q.; and Huang, X.-J. 2020. Tasty Burgers, Soggy Fries: Probing Aspect Robustness in Aspect-Based Sentiment Analysis. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 3594–3605.

Xu, K.; Ba, J.; Kiros, R.; Cho, K.; Courville, A.; Salakhudinov, R.; Zemel, R.; and Bengio, Y. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, 2048–2057. PMLR.

Xu, W.; Liu, Q.; Wu, S.; and Wang, L. 2023. Counterfactual Debiasing for Fact Verification. In Rogers, A.; Boyd-Graber, J.; and Okazaki, N., eds., *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 6777–6789. Toronto, Canada: Association for Computational Linguistics.

Yang, X.; Peng, Y.; Ma, H.; Xu, S.; Zhang, C.; Han, Y.; and Zhang, H. 2024. Lever LM: configuring in-context sequence to lever large vision language models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Yang, Z.; Qi, P.; Zhang, S.; Bengio, Y.; Cohen, W.; Salakhutdinov, R.; and Manning, C. D. 2018. HotpotQA: A Dataset for Diverse, Explainable Multi-hop Question Answering. In Riloff, E.; Chiang, D.; Hockenmaier, J.; and Tsujii, J., eds., *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 2369–2380. Brussels, Belgium: Association for Computational Linguistics.

Ye, J.; Chen, X.; Xu, N.; Zu, C.; Shao, Z.; Liu, S.; Cui, Y.; Zhou, Z.; Gong, C.; Shen, Y.; et al. 2023. A comprehensive capability analysis of gpt-3 and gpt-3.5 series models. *arXiv preprint arXiv:2303.10420*.

Zhang, C.; Zhang, L.; and Zhou, D. 2024. Causal Walk: Debiasing Multi-Hop Fact Verification with Front-Door Adjustment. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 19533–19541.

Zhang, L.; Zhang, C.; and Zhou, D. 2023. Multi-Relational Probabilistic Event Representation Learning via Projected Gaussian Embedding. In *Findings of the Association for Computational Linguistics: ACL 2023*, 6162–6174.

Zhang, S.; Pan, L.; Zhao, J.; and Wang, W. Y. 2023. Mitigating language model hallucination with interactive question-knowledge alignment. *arXiv preprint arXiv:2305.13669*.

Zhao, Z.; Wallace, E.; Feng, S.; Klein, D.; and Singh, S. 2021. Calibrate before use: Improving few-shot performance of language models. In *International Conference on Machine Learning*, 12697–12706. PMLR.

Zhou, D.; Schärli, N.; Hou, L.; Wei, J.; Scales, N.; Wang, X.; Schuurmans, D.; Cui, C.; Bousquet, O.; Le, Q.; et al. 2022. Least-to-most prompting enables complex reasoning in large language models. *arXiv preprint arXiv:2205.10625*.

Zhu, J.; Wu, S.; Zhang, X.; Hou, Y.; and Feng, Z. 2023. Causal Intervention for Mitigating Name Bias in Machine Reading Comprehension. In *Findings of the Association for Computational Linguistics: ACL 2023*, 12837–12852.