

# CausalFlip: A Benchmark for LLM Causal Judgment Beyond Semantic Matching

Yuzhe Wang  
The University of Virginia  
Charlottesville, USA  
srq3da@virginia.edu

Yaochen Zhu  
The University of Virginia  
Charlottesville, USA  
uqp4qh@virginia.edu

Jundong Li  
The University of Virginia  
Charlottesville, USA  
jl6qk@virginia.edu

## Abstract

As large language models (LLMs) witness increasing deployment in complex, high-stakes decision-making scenarios, it becomes imperative to ground their reasoning in causality rather than spurious correlations. However, strong performance on traditional reasoning benchmarks does not guarantee true causal reasoning ability of LLMs, as high accuracy may still arise from memorizing semantic patterns instead of analyzing the underlying true causal structures. To bridge this critical gap, we propose a new causal reasoning benchmark, **CausalFlip**, designed to encourage the development of new LLM paradigm or training algorithms that ground LLM reasoning in causality rather than semantic correlation. CausalFlip consists of causal judgment questions built over event triples that could form different confounder, chain, and collider relations. Based on this, for each event triple, we construct pairs of semantically similar questions that reuse the same events but yield opposite causal answers, where models that rely heavily on semantic matching are systematically driven toward incorrect predictions. To further probe models' reliance on semantic patterns, we introduce a noisy-prefix evaluation that prepends causally irrelevant text before intermediate causal reasoning steps without altering the underlying causal relations or the logic of the reasoning process. We evaluate LLMs under multiple training paradigms, including answer-only training, explicit Chain-of-Thought (CoT) supervision, and a proposed internalized causal reasoning approach that aims to mitigate explicit reliance on correlation in the reasoning process. Our results show that explicit CoT can still be misled by spurious semantic correlations, where internalizing reasoning steps yields substantially improved causal grounding, suggesting that it is promising to better elicit the latent causal reasoning capabilities of base LLMs. The code and benchmark is available at <https://github.com/Yuzhe-W/CausalFlip>.

## CCS Concepts

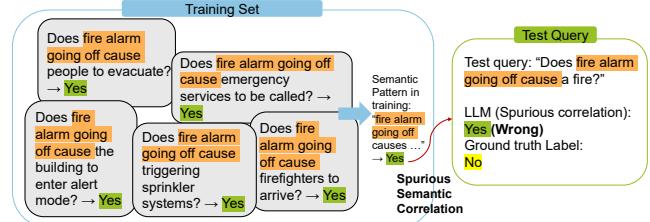
- Computing methodologies → Natural language processing; Causal reasoning and diagnostics.

## Keywords

Large Language Models, Causal Reasoning, Benchmarks, Datasets

## 1 Introduction

With the unprecedented knowledge and reasoning ability of large language models (LLM), recent years have witnessed an increased deployment in critical domains, such as medical diagnosis [23, 27], financial analysis [35, 36], and legal systems [16]. However, these LLMs are often questioned in terms of their reliability in reasoning and decisions, where hallucinations could cause devastating outcomes [12, 19, 21]. Traditional auto-regressive LLMs [2, 6, 24]



**Figure 1: One representative example where training samples may create a spurious semantic correlation with a wrong answer that leads LLM to an incorrect causal judgment.**

optimize the model by predicting the next token *conditioning on* the context, which essentially leverage semantic patterns that correlate with the training data to reason and answer for new questions [17, 31] (see Fig. 1). In addition, traditional reasoning benchmarks [18, 25, 28, 30] may not reliably reveal the causal reasoning limitations of LLMs, as models can often achieve high scores by exploiting semantic correlations rather than engaging in deeper causal reasoning. Therefore, a significant gap exists between traditional benchmarks and the causal reasoning ability of LLMs, which severely hinders the development of new LLM paradigms or training algorithms that fundamentally ground its reasoning in causality.

In the traditional paradigm of auto-regressive LLMs, chain-of-thought (CoT) prompting may appear as the closest form of strategy to improve LLM performance on causal reasoning tasks [14, 34], where the latter reasoning steps and the final answer "causally" depend on the previous reasoning steps. On traditional reasoning benchmarks, explicitly encouraging the LLM to generate intermediate reasoning steps often improves accuracy and makes predictions easier to inspect. However, explicit CoT leads to higher latency and token usage [4, 26]. Furthermore, in essence, the CoT structures themselves can be viewed as modeling the conditional distribution of the next reasoning steps based on the context of the question and previous reasoning steps, which still rely on semantic patterns for the model to memorize and not causal. Recent work of implicit CoT proposes a different approach [5]: for math tasks like multi-digit multiplication, the method gradually removes CoT steps during the model's training, nudging the model to encode the CoT in its internal weights, which allows the LLMs to solve these math tasks in high accuracy. However, it remains unclear whether such internalization better encourages the model to internalize causal reasoning model weights during the forward propagation.

In this paper, we pioneer to bridge the gap by establishing a novel benchmark, i.e., **CausalFlip**, aiming to encourage the development of new LLM paradigms or training algorithms that fundamentally

ground their reasoning in causality. The overarching principle is that LLMs that are designed or trained to generate answers based on semantic similarity (e.g., with auto-regressive next token prediction) will suffer from spurious semantic correlation, i.e., the correlation between the semantics in the question and the *wrong* causal answer, whereas the models that can truly leverage causality to reason prevail. Shown in top of Figure 2, our benchmark has three sub-datasets, i.e., **confounder**, **chain**, and **collider**. Each sub-dataset includes causal questions with two causal structures: *base* and *opposite*. Specifically, *Base* refers to the case where the event in the question contains a causal structure consistent with the sub-dataset name (e.g., in the confounder dataset, the base causal structure refers to questions where included events form the confounder structure), whereas *Opposite* denotes the question with events that form different causal structure. For each structure, we collected paired causal inquiries, where two semantically similar questions are created based on the same events with different labels (Although the event set could be different for the Base/Opposite structure). As the *base* pair example shown in Figure 2’s **blue part**: Q1 and Q2 form two types of causal questions with the same event triple (Umbrella sales, Traffic jams, Monsoon season), and they have semantically similar fixed templates, specified in the caption of Figure 2. From the *base* causal structure (confounder), the pairwise questions have different labels. Therefore, when we split Q1/Q2 to the training/test set, it induces spurious semantic correlation between the question semantic and label where LLMs that rely on semantic correlation will fail. For pairs in the *opposite* structure (Figure 2’s **orange part**), Q1 and Q2 ask the same two types of causal questions over different event triples (Typing practice, Typing speed, City budget week), and use the same templates as in the *base* question pair. But under the *opposite* causal structure, the question that has the same template with the *base* structure has the opposite answer. This prevents possible positive spurious correlations (i.e., the semantic patterns in the question that correlate with the correct label) from only including the *base* structure in each sub-dataset, where Q1’s labels are always “No” and Q2’s labels are always “Yes”, and the model could rely on the specific question template to predict the label.

To further reduce possible positive spurious correlation of answer on any phrasing pattern in the inquiries, two templates, *Default* (default phrasing) and *Alternative* (an alternative phrasing grounded in the same causal structure) are provided, which leads to four categories: Base–Default (BD), Base–Alternative (BA), Opposite–Default (OD), and Opposite–Alternative (OA). Each category has the same number of causal question pairs. Finally, we use a pairwise train-test split: for each pair, one question is used for training, and its semantically similar counterpart with the opposite label is held out for evaluation, so approaches that rely heavily on semantic pattern matching are systematically penalized, and strong performance needs to be grounded in causal structure.

As an initial exploration of causality-grounded LLM, we propose ***implicit causal reasoning***, a novel LLM training strategy that internalizes causal reasoning steps by progressively masking the supervised intermediate causal reasoning step tokens, as a strong baseline on the **CausalFlip** dataset. Under the progressive token mask, the model is encouraged to encode the causal reasoning in its parameters rather than depending on explicit generation

steps. To further investigate whether implicit causal reasoning encourages the model to focus more on causal reasoning rather than semantics, we further extend the **CausalFlip** dataset by prepending a generic, causally irrelevant prefix before the CoT as semantic noise, while keeping the CoT content and the underlying causal structure unchanged. If explicit CoT relies on semantic patterns instead of causality for reasoning, it should suffer more from the injected noise. In contrast, implicit CoT’s token-removal schedule may teach the model to “ignore” such prefixes, where the explicit spurious correlation on generated reasoning gaps can be alleviated.

The extensive experiments conducted on the **CausalFlip** benchmark highlight two main findings. First, the limited performance of the standard LLM training strategy (no-CoT) underscores that success on CausalFlip requires grounding predictions in causal structure, rather than relying on spurious semantic correlations. In addition, the supervision of intermediate causal reasoning steps (explicit-CoT and implicit causal reasoning) improves accuracy on causal tasks compared to the pretraining strategy baseline and no-CoT fine-tuning. This suggests the intermediate causal reasoning steps as a main driver of better causal-judgment performance on CausalFlip, indicating that answering these questions needs reasoning grounded in causal structure. Second, when a fixed noisy prefix is added without changing the underlying causal structure, explicit-CoT exhibits a larger performance degradation, while implicit causal reasoning demonstrates robustness and achieves higher accuracy than explicit-CoT. This suggests that internalizing the reasoning process can reduce reliance on spurious semantic correlations on explicitly generated intermediate causal reasoning steps and better preserve decisions to causal questions. The contribution of this work can be summarized three-fold as:

- We introduce **CausalFlip**, a causality-grounding benchmark consisting of paired, semantically similar questions that involve the same events but receive opposite labels. This design aims to stimulate the development and training of LLMs whose reasoning is explicitly anchored in causal structure rather than superficial semantic cues.
- We propose an *implicit causal reasoning* training strategy that progressively weakens supervision on intermediate reasoning tokens to encourage internalization, and we show it is consistently more robust than explicit-CoT under noisy-prefix distraction, indicating reduced dependence on spurious semantic correlations towards causal tasks.
- We introduce a novel *noisy-prefix evaluation* to test whether training strategies encourage causal reasoning rather than relying on semantic pattern matching as with traditional auto-regressive language modeling.

## 2 Preliminaries

### 2.1 LLM Basics

We follow the standard language modeling formulation and view LLM as a parameterized probabilistic model over sequences of discrete tokens [1, 2, 24]. Let  $\mathcal{V}$  denote the vocabulary set of the LLM and let a tokenized sequence be  $\mathbf{x} = (x_1, \dots, x_T)$  with  $x_t \in \mathcal{V}$ . An autoregressive LLM with parameters  $\theta$  defines a joint distribution

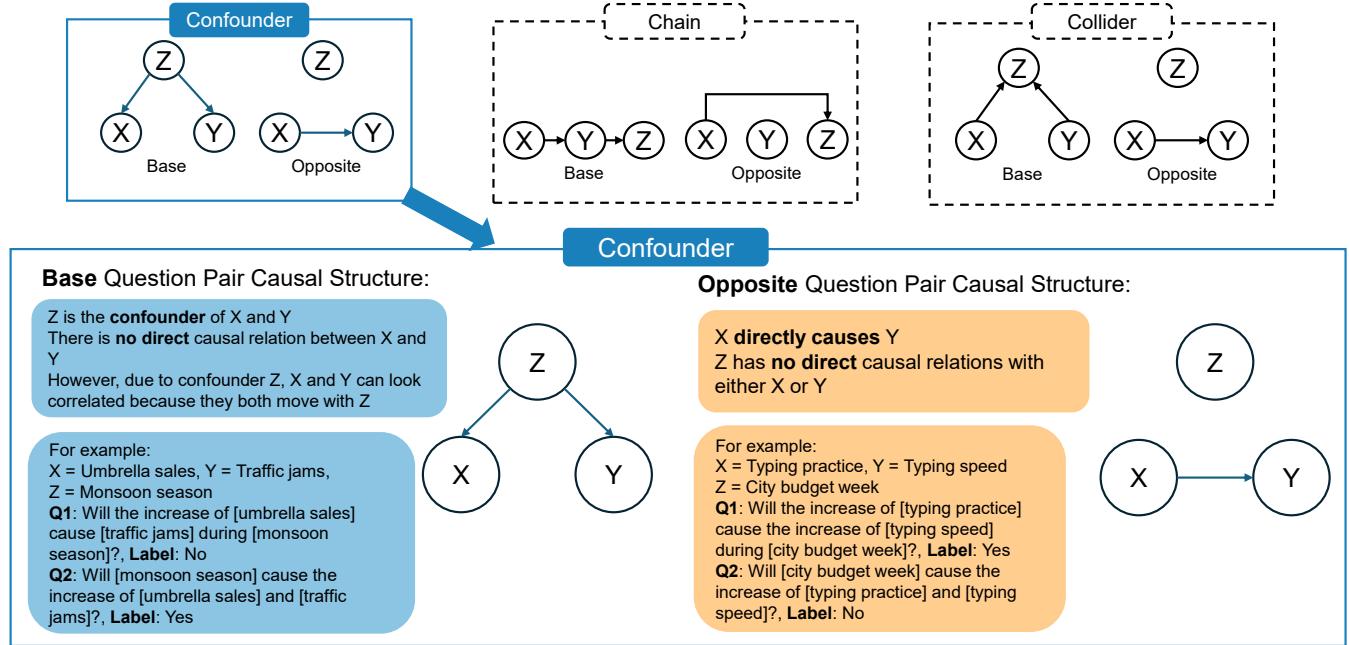


Figure 2: Overview of the causal structures used in our benchmark and an example of base, opposite pairs. The top row shows the causal structures of three sub-datasets (confounder, chain, collider); The bottom expands the confounder case and highlights a base vs. opposite question pairs: In both base and opposite question pairs, Q1 asks whether the causal relation from X to Y exists under the context of Z, with fixed template: **Will the increase of X cause Y during Z?**, and Q2 asks whether the causal relations between Z and X / Y exists, with fixed template: **Will Z cause the increase of X and Y**.

over  $\mathbf{x}$  via the chain-rule factorization as follows:

$$p_{\theta}(\mathbf{x}) = \prod_{t=1}^T p_{\theta}(x_t | x_{<t}), \quad (1)$$

where  $x_{<t} = (x_1, \dots, x_{t-1})$  is the prefix context. Autoregressive generation produces tokens sequentially from left to right by decoding  $x_t \sim p_{\theta}(\cdot | x_{<t})$ . In practice,  $p_{\theta}(x_t | x_{<t})$  is commonly instantiated by a Transformer architecture based on self-attention [32]. Concretely, let  $h_t$  denote the final-layer hidden representation at the last position of the prefix  $x_{<t}$ . The next-token distribution is then obtained by applying a softmax over vocabulary logits,

$$p_{\theta}(x_t = v | x_{<t}) = \frac{\exp(s_{\theta}(v; h_t))}{\sum_{v' \in \mathcal{V}} \exp(s_{\theta}(v'; h_t))}, \quad (2)$$

where  $s_{\theta}(\cdot; h_t)$  denotes the score (logit) assigned to each vocabulary token. In generation, the model conditions on a given context  $\mathbf{c} = (c_1, \dots, c_K)$  and generates a continuation  $\mathbf{y} = (y_1, \dots, y_M)$ . The induced conditional distribution follows the same autoregression:

$$p_{\theta}(\mathbf{y} | \mathbf{c}) = \prod_{m=1}^M p_{\theta}(y_m | \mathbf{c}, y_{<m}), \quad (3)$$

where  $y_{<m} = (y_1, \dots, y_{m-1})$ . The standard training objective of LLMs is the maximum likelihood estimation over a corpus  $\mathcal{D}$  (equivalently, minimizing token-level cross-entropy):

$$\max_{\theta} \sum_{\mathbf{x} \in \mathcal{D}} \sum_{t=1}^T \log p_{\theta}(x_t | x_{<t}). \quad (4)$$

This next-token prediction objective rewards any statistical regularities in the context that improve predictive likelihood, which can include spurious semantic correlations with downstream labels [2].

At inference time, a completion is generated by decoding from Eq. (3), for example via greedy decoding, or stochastic sampling [8, 29]. In particular, sampling-based decoding, e.g., top- $k$ , is commonly used to trade off diversity and coherence in the generated text [7]. From Eq. (3) we can find that, the training of LLMs with autoregressive language modeling relies on conditional distribution instead of causal analysis on the questions, which is susceptible to provide the wrong answers based on spurious semantic correlations.

## 2.2 Chain-of-Thought Reasoning

In this part, we formalize explicit Chain-of-Thought (CoT), which provides the foundation for the proposed implicit causal reasoning and the robustness evaluation. CoT reasoning refers to generating intermediate natural-language reasoning steps before producing a final answer, which has been shown to improve performance on multi-step reasoning tasks [14, 34]. Formally, let the input/prompt be  $\mathbf{x}$ . Under CoT, the model's output is a single sequence  $\mathbf{z}$ , which we split into (i) intermediate reasoning steps (CoT) tokens  $\mathbf{s}$  and (ii) final answer tokens  $\mathbf{y}$  as  $\mathbf{z} = [\mathbf{s}; \mathbf{y}]$ . An autoregressive language model defines the conditional likelihood of the full output as

$$p_{\theta}(\mathbf{z} | \mathbf{x}) = \prod_{t=1}^{|\mathbf{z}|} p_{\theta}(z_t | \mathbf{x}, z_{<t}), \quad (5)$$

where  $\mathbf{z}_{\prec t}$  is the prefix of previously generated output tokens. By viewing  $\mathbf{z}$  as  $\mathbf{z} = [\mathbf{s}; \mathbf{y}]$ , it is equivalent to:

$$p_\theta(\mathbf{s}, \mathbf{y} | \mathbf{x}) = p_\theta(\mathbf{s} | \mathbf{x}) p_\theta(\mathbf{y} | \mathbf{x}, \mathbf{s}). \quad (6)$$

From this perspective,  $\mathbf{s}$  serves as an intermediate reasoning steps which the final decision  $\mathbf{y}$  is conditioned on [34]. At inference time, CoT typically proceeds by generating a *single* reasoning steps' rollout  $\hat{\mathbf{s}}$  via a decoding procedure  $\text{Dec}(\cdot)$  (e.g., greedy decoding or sampling), then predicting the answer  $\hat{\mathbf{y}}$  conditioned on it:

$$\hat{\mathbf{s}} = \text{Dec}_\theta(\mathbf{x}), \quad \hat{\mathbf{y}} = \text{Dec}_\theta([\mathbf{x}; \hat{\mathbf{s}}]). \quad (7)$$

However, CoT is essentially correlational instead of causal, as each intermediate step is still based on the conditional distribution given the question and previously generated reasoning steps, which therefore relies on spurious semantic correlation instead of grounding the reasoning in the true underlying causal structure.

### 2.3 Problem Formulation

To address the limitation of traditional auto-regressive LLM and explicit CoT, we study causal-judgment question answering with LLMs. Let  $Q$  denote the space of natural-language causal questions, and let  $\mathcal{A}$  denote the answer space, where  $\mathcal{A} = \{\text{Yes}, \text{No}\}$  denotes the binary causal answers. We assume each question  $q \in Q$  is associated with an underlying causal graph  $G$  and a target causal label  $a \in \mathcal{A}$  that reflects the correct causal judgment under  $G$ . We represent the benchmark as a dataset  $\mathcal{D} = \{(q_i, G_i, a_i)\}_{i=1}^N$ . Here, we assume a general formulation of LLM with parameters  $\theta$ , which defines the probability of the answer (in textual format) given the question  $p_\theta(a | q)$ ,  $a \in \mathcal{A}$ , and predicts an answer by  $\hat{a}(q) = \arg \max_{a \in \mathcal{A}} p_\theta(a | q)$ .

## 3 Benchmark Design

The aim of this benchmark is to address the limitations of traditional benchmarks in comprehensively evaluating LLMs' ability to make reliable causal judgments. CausalFlip is built from event triples  $(X, Y, Z)$  and formulates binary causal judgment questions across three sub-datasets: *confounder*, *chain*, and *collider*. For each sub-dataset, we define *Base* and *Opposite* causal structures and construct semantically similar question pairs whose correct labels flip according to the underlying causal structure. Combined with pairwise train-test split, this design penalizes models from relying on spurious semantic correlations for predictions. Furthermore, we control question templates through Default and Alternative templates to reduce template-driven positive spurious correlation.

### 3.1 Semantically Similar, Label-Flipped Pairs with Pairwise Train–Test Split

The overarching principle for developing the **CausalFlip** dataset is to introduce *paired questions with spurious correlations in semantics*, i.e., *questions with similar semantic meanings but lead to different causal answers*. The two questions differ primarily in the causal relation they query, and therefore have opposite labels, corresponding to the two question types *(i)*, *(ii)* defined in subsection 3.2. For example, in the confounder dataset, a pair contrasts a direct-effect question, *Will the increase of  $X$  cause  $Y$  during  $Z$ ?*, with a semantically similar confounder question, *Will  $Z$  cause the*

*increase of  $X$  and  $Y$ ?*. For evaluation, we adopt a *pairwise* train–test split. In each dataset, for each pair, we assign one question to the training set and the other to the test set while keeping the number of Q1/Q2 balanced across the training and test splits. This ensures that every test question has a counterpart in training that shares the same event triples and similar semantics but carries the opposite label. Consequently, if the model relies heavily on semantic patterns, it will predict the same label as the corresponding training instance, thereby failing on the paired testing question where the correct label is flipped. Thus, a high performance on this benchmark requires the understanding of causal structures, rather than relying on semantic matching.

### 3.2 Causal Structures and Induced Questions

The established **CausalFlip** benchmark is composed of three sub-datasets, i.e., *confounder*, *chain*, and *collider*, where each dataset type studies the canonical causal structure among  $(X, Y, Z)$ . Specifically, within each dataset type, we define two causal structures: *Base* (*B*) and *Opposite* (*O*). *Base* follows the canonical causal graph of the dataset type, while *Opposite* uses an alternative causal graph that is constructed to yield the opposite labels to the same paired question types compared to the *Base*. With this design, the model cannot rely on the positive spurious correlation between the question template and the label (e.g., with only *Base* pairs, question 1's label is always "No" and question 2's is always Yes). This forces correct predictions to depend on the underlying causal structures. To keep the benchmark balanced, we also construct the same number of pairs under the *Base* and *Opposite* causal structures in each dataset.

**3.2.1 Confounder Dataset.** **Base (B)** represents the confounder *base* structure, where  $Z \rightarrow X$  and  $Z \rightarrow Y$ , with no directed causal path from  $X$  to  $Y$ . Each question pair in this dataset includes two types of questions: *(i)* is a direct-effect question asking whether increasing  $X$  causes  $Y$  in the context of  $Z$  (we include  $Z$  as context to keep this question semantically close to question *(ii)*), and *(ii)* is a confounder question asking whether  $Z$  causes both  $X$  and  $Y$ . Under the *Base* structure, Question *(i)* is labeled "No" and Question *(ii)* is labeled "Yes". **Opposite (O)** represents the *opposite* structure under confounder dataset, that follows  $X \rightarrow Y$ , and there is no causal path from  $Z$  to both  $X$  and  $Y$ , which means  $Z$  is non-causal w.r.t. both  $X$  and  $Y$ . The question types *i*, *ii* remain the same, but due to the opposite causal structure, the labels flip: Question *(i)* is assigned Yes and Question *(ii)* is assigned No.

**3.2.2 Chain Dataset.** **Base (B)** represents the chain *base* structure follows  $X \rightarrow Y \rightarrow Z$ , with no direct causal path  $X \rightarrow Z$ , which means any effect of  $X$  on  $Z$  is mediated through  $Y$ . Each question pair in this dataset includes two question types: *(i)* is a direct-effect question asking whether increasing  $X$  directly causes  $Z$  in the context of  $Y$  (we include  $Y$  as context to keep this question semantically close to question *(ii)*), and *(ii)* is a mediated path question asking whether increasing  $X$  increases  $Y$ , which in turn increases  $Z$ . Under *Base* structure, Question *(i)* is labeled No and Question *(ii)* is labeled Yes. **Opposite (O)** represents *opposite* structure under chain dataset, follows  $X \rightarrow Z$ , and there is no causal path from  $X$  to  $Y$  and  $Y$  to  $Z$ . The question types remain the same,

but due to the opposite causal structure, labels flip: Question **(i)** is assigned Yes and Question **(ii)** is assigned No.

**3.2.3 Collider Dataset.** **Base (B)** represents the collider base structure follows  $X \rightarrow Z$  and  $Y \rightarrow Z$ , with no directed path between  $X$  and  $Y$ ;  $Z$  is a collider with two independent parents. Each instance includes two question types: **(i)** is a direct-effect question asking whether increasing  $X$  causes  $Y$  *in the context of*  $Z$  (we include  $Z$  as context to keep this question semantically close to Question **(ii)**), and **(ii)** is a collider question asking whether increasing  $X$  causes  $Z$  and increasing  $Y$  causes  $Z$ . Under Base structure, Question **(i)** is labeled No and Question **(ii)** is labeled Yes. **Opposite (O)** represents the *opposite* structure under collider dataset, follows  $X \rightarrow Y$ , and there is no causal path from  $X$  to  $Z$  and no causal path from  $Y$  to  $Z$ . The question types remain the same, but labels flip: Question **(i)** is assigned Yes and Question **(ii)** is assigned No.

### 3.3 Question Templates

The design of pairwise split introduces a strong spurious (negative) correlation between question semantics and causal answers, which ideally should lead to the failure of LLM training strategies that purely rely on semantic correlation to generate answers. However, we find that the auto-regressive LLMs may still exploit shortcuts tied to phrase patterns in the question, i.e., certain remaining *positive* semantics correlation with the answer. To further reduce such shortcuts on wording style, we construct two question phrasing templates as randomization to reduce the positive spurious correlation: **Default (D)** and **Alternative (A)**.

- **Default (D)** is default question phrasing in **CausalFlip**. It expresses each dataset's two question types as direct binary questions (e.g., Will the increase in  $X$  cause  $Y$  during  $Z$  . . . ?). For each dataset type confounder, chain, collider, we generate a set of paired questions using the Default templates.
- **Alternative (A)** is a different phrasing to express the same types of causal questions. It is a declarative form (e.g., An increase in  $X$  leads to an increase in  $Y$  . . . ), and the model judges whether the statement is correct with a binary {Yes, No} answer. For each dataset type, we generate a separate set of paired questions using the Alternative templates. Within each dataset, we balance the samples across template types by ensuring the same number of Default pairs and Alternative pairs. Combining template family (Default vs Alternative) with causal structure (Base vs Opposite) yields four categories: **BD**, **BA**, **OD**, and **OA**.

Since each dataset contains the same number of Base and Opposite pairs and the same number of Default and Alternative pairs, the number of samples in BD, BA, OD, and OA are matched. This balance prevents any structure-template category (BD/BA/OD/OA) from being over-represented, and helps ensure that predictions are grounded in causal structure rather than template shortcuts.

## 4 Implicit Causal Reasoning

In the previous section, we have proposed **CausalFlip** benchmark that introduces spurious semantic correlation between *causal question semantics* and *labels* such that model trained based on *semantic*

*pattern matching* fails to encourage the development of new LLM *training strategies* to reason on *causal structures*. In this section, we propose a new strategy, implicit causal reasoning, to reduce the explicit-CoT's reliance on spurious and explicit semantic correlations in causal questions.

### 4.1 Motivation and Overview

**CausalFlip** is designed to penalize the performance of models that rely on semantics for prediction. This motivates training strategies that reduce a model's tendency to rely on semantic correlation when making causal judgments. As a preliminary exploration in fundamentally grounding the LLM reasoning in causality, we adopt a *progressive causal reasoning steps mask* strategy, where we progressively mask the causal reasoning steps, removing them from the training loss to specifically mitigate its reliance on spurious semantic correlations in explicitly generated causal reasoning steps and encourage generations grounded in causal structure.

### 4.2 Progressive Causal Reasoning Steps Mask

In this part, we consider training samples consisting of an input question  $\mathbf{x}$ , a binary answer  $y \in \{\text{Yes}, \text{No}\}$ , and explicit causal reasoning steps  $s = (s_1, \dots, s_L)$ . In our setting,  $s$  is the intermediate causal reasoning steps derived from the corresponding causal graph, which states the key structural conditions and leads to the answer.

**4.2.1 Explicit Causal Reasoning Steps Supervision.** We start from explicit CoT, where all causal reasoning steps' tokens are included in training loss. The standard autoregressive next-token objective is applied on all intermediate causal reasoning steps' tokens  $s = (s_1, \dots, s_L)$  as well as the final answer  $y$  as:

$$\mathcal{L}_{\text{explicit}}(\theta) = - \sum_{(\mathbf{x}, s, y) \in \mathcal{D}} \log p_\theta(s, y | \mathbf{x}), \quad (8)$$

Equivalently, the token-wise likelihood is given by Eq. (9). Here  $L$  denotes the number of tokens in the causal reasoning steps, and  $k \in \{1, \dots, L\}$  indexes a token position within  $s$ :

$$\mathcal{L}_{\text{explicit}}(\theta) = - \sum_{(\mathbf{x}, s, y) \in \mathcal{D}} \left( \sum_{k=1}^L \log p_\theta(s_k | \mathbf{x}, s_{<k}) + \log p_\theta(y | \mathbf{x}, s) \right). \quad (9)$$

**4.2.2 Progressive Causal Reasoning Steps Removal.** For implicit causal reasoning, we progressively mask the earliest causal reasoning steps' tokens from supervision. At training step  $t$ , we remove the first  $r(t)$  tokens from supervision and only compute the loss on the remaining part. Formally, we apply a step-based mask function  $m_k(t) \in \{0, 1\}$ , over causal reasoning steps token positions  $k$  as follows:

$$\begin{aligned} \mathcal{L}_{\text{mask}}(\theta; t) = & - \sum_{(\mathbf{x}, s, y) \in \mathcal{D}} \left( \sum_{k=1}^L m_k(t) \log p_\theta(s_k | \mathbf{x}, s_{<k}) \right. \\ & \left. + \log p_\theta(y | \mathbf{x}, s) \right), \quad m_k(t) = 1[k > r(t)]. \end{aligned} \quad (10)$$

where  $m_k(t) = 0$  indicates that the loss on  $s_k$  is masked out, and  $m_k(t) = 1$  indicates standard supervision (Eq. (10)). Thus, the first  $r(t)$  causal reasoning steps' tokens  $\{s_1, \dots, s_{r(t)}\}$  receive no direct

supervision, while the remaining suffix  $s_{(t)+1:L}$  is still included in the loss. We always compute training loss on the final answer  $y$  to preserve task performance as explicit causal reasoning steps supervision diminishes.

## 5 Empirical Study

In this section, we evaluate four strategies (i.e., naive pretraining strategy, no-CoT, explicit-CoT, and implicit causal reasoning fine-tuning) on our **CausalFlip** benchmark. We first find that the naive pretraining and no-CoT strategies perform poorly, confirming that LLMs need to ground their predictions in causal structure instead of semantic patterns to succeed in CausalFlip benchmark. In addition, we find that training strategies with intermediate causal reasoning steps achieve much stronger causal-judgment performance. Finally, we further demonstrate that when injecting a noisy semantic prefix, implicit causal reasoning maintains higher robustness than explicit-CoT, indicating its less reliance on spurious semantic correlations and a reliable grounding in causal structure.

### 5.1 Research Questions

**RQ1: How do no-CoT, explicit-CoT, and implicit causal reasoning training strategies perform on our causal benchmark compared to the naive pretraining strategy?**

This question compares strategies in terms of their ability to make *causal* judgments when spurious semantic correlations are not a reliable shortcut, due to the pair-wise split design in **CausalFlip**. All strategies use the same train/test split and the same evaluation protocol and we report overall accuracy for each model.

**RQ2: Could the implicit causal reasoning model perform more robust in causal reasoning tasks than explicit-CoT by reducing its reliance on semantic patterns?**

As we analyze in subsection 2.1, autoregressive LLMs are optimized to fit conditional distributions, and thus can exploit positive spurious semantic correlations to make predictions, instead of grounding in the underlying causal structure [17, 31]. Motivated by this, we further test whether implicit causal reasoning reduces reliance on semantic patterns by comparing it with explicit-CoT under our noisy-prefix evaluation. Specifically, we inject the same causally irrelevant prefix before intermediate causal reasoning steps and measure how each model’s accuracy changes, using performance degradation as an indicator of reliance on semantic patterns versus causal structure.

### 5.2 Baseline and Training Strategies

We compare four strategies to answer the two research questions raised in this paper. Across all experiments, we use **Llama-3.2-3B-Instruct** as the base model.

- **Naive pretraining strategy (No Fine-tuning):** This is the base model without any training on our dataset **CausalFlip**. It measures the baseline pretraining’s causal-judgment performance under pretraining alone and serves as a reference for evaluating fine-tuning strategies (no-CoT, explicit-CoT, and implicit causal reasoning).
- **No-CoT fine-tuning:** This is the standard answer-only supervised fine-tuning baseline, where the model is trained

to directly predict the final answer given a causal question from **CausalFlip**. In this setting, no intermediate causal reasoning steps are provided during training, so the supervision is applied only to the final Yes/No decision.

- **Explicit-CoT fine-tuning:** Different from no-CoT model, this model is trained with the full intermediate causal reasoning steps followed by the final answer, and supervision is applied to both the intermediate causal reasoning tokens and the final answer.
- **Implicit causal reasoning fine-tuning:** We progressively mask an increasing prefix of intermediate reasoning tokens from the training loss while always supervising the final decision. This shifts prediction’s reliance from explicit semantics to the underlying causal structure.

These four training strategies enable the evaluation of how different supervision strategies influence model performance on our proposed causal question dataset.

### 5.3 Training and Evaluation Setups

During training and evaluation, all strategies use the same fixed train/test split set with the same causal questions defined in Section 3 to ensure fair comparisons. In the following part, we introduce the training protocol and evaluation protocol in more detail.

**5.3.1 Training.** Each fine-tuning instance is composed of (i) the causal question from **CausalFlip** as defined earlier in Section 3, and (ii) the answer of the question, which is the binary answer in {Yes, No}. For *no-CoT*, the instance is same as we stated before, that is the question followed by the answer. For *explicit-CoT* and *implicit causal reasoning*, we include the intermediate causal reasoning steps placed between the causal question and the answer (see subsection 4.2). For example, for *base* causal structure pairs in the *confounder* dataset, we employ the following causal reasoning step:

No directed causal path from X to Y AND adjusting for Z closes the backdoor between X and Y, therefore

The final answer follows immediately.

**5.3.2 Evaluation.** At test time, models are presented with a question-only prompt and evaluated based on their generated responses. Each prompt contains two parts: (i) a *format instruction* that constrains the final answer within the model’s output as a binary decision {Yes, No}, which makes it amenable to parse and evaluate, even in the settings of explicit-CoT and implicit causal reasoning, where intermediate causal reasoning steps are generated before the final answer; and (ii) a causal question from **CausalFlip**, defined in section 3. For each prompt, the model generates a single response that we extract the final answer and use it to compute task accuracy.

**5.3.3 Metrics.** We score the generation based only on the final Yes/No answer without judging any intermediate causal reasoning steps. Specifically, for each test example, we take the model’s final answer, which is the predicted label  $\hat{y}_i$ , and compare it with the ground-truth label  $y_i$ . Accuracy is computed as:

$$\text{Accuracy} = \frac{1}{N} \sum_{i=1}^N \mathbf{1}(\hat{y}_i = y_i). \quad (11)$$

**Table 1: Performance comparisons on Confounder Dataset**

Model	Accuracy	Correct / Total	Valid Yes/No
Naive pretraining strategy	0.529	529 / 1000	1000 / 1000
No CoT	0.524	524 / 1000	1000 / 1000
Explicit CoT	0.892	892 / 1000	987 / 1000
Implicit Causal Reasoning	0.900	900 / 1000	997 / 1000

**Table 2: Performance comparisons on Chain Dataset**

Model	Accuracy	Correct / Total	Valid Yes/No
Naive pretraining strategy	0.612	612 / 1000	1000 / 1000
No CoT	0.639	639 / 1000	1000 / 1000
Explicit CoT	0.690	690 / 1000	989 / 1000
Implicit Causal Reasoning	0.757	757 / 1000	998 / 1000

**Table 3: Performance comparisons on Collider Dataset**

Model	Accuracy	Correct / Total	Valid Yes/No
Naive pretraining strategy	0.629	629 / 1000	1000 / 1000
No CoT	0.655	655 / 1000	1000 / 1000
Explicit CoT	0.856	856 / 1000	1000 / 1000
Implicit Causal Reasoning	0.849	849 / 1000	999 / 1000

where  $N$  is the number of test examples,  $y_i \in \{\text{Yes}, \text{No}\}$  is the ground-truth label,  $\hat{y}_i \in \{\text{Yes}, \text{No}\}$  is the model’s predicted label (final answer), and  $1[\cdot]$  is the indicator function. To measure the performance across the models, we report accuracy on clean inputs and under the noisy-prefix setting described below.

**5.3.4 Noisy-Prefix Evaluation.** To test whether the model focuses on causal reasoning or semantics, we inject the noisy prefix before the intermediate causal reasoning steps in the training samples. The prefix is a fixed chunk of natural-language sentences designed to serve as a semantic distraction while remaining causally irrelevant, which does not violate the intermediate reasoning process, nor change the underlying causal structure. We apply the same prefix to all questions to ensure consistency. This also avoids spurious correlations between different noisy prefixes’ semantics and labels.

We compare explicit-CoT and implicit causal reasoning because they represented two training strategies for leveraging intermediate causal reasoning. Explicit-CoT exposes the model to full intermediate causal reasoning steps during training, which may increase reliance on spurious semantic correlations and make its final decision more sensitive to the noisy prefix. In contrast, implicit causal reasoning progressively masks causal reasoning steps tokens over training. We evaluate both strategies under the noisy-prefix.

#### 5.4 Performance on the CausalFlip (RQ1)

We first answer RQ1 by comparing four strategies’ performances on CausalFlip: the naive pretraining strategy, standard answer-only fine-tuning (**no-CoT**), **explicit-CoT** fine-tuning, and **implicit causal reasoning** fine-tuning. We report accuracy on clean (without noisy prefix) inputs for **Confounder**, **Chain**, and **Collider** datasets (Table 1, 2, 3)

On all three datasets, we observe low performance of the naive pretraining and no-CoT fine-tuning strategies, as these strategies fundamentally rely on semantics correlations rather than grounding in causal structure to train the LLMs. On Confounder, naive/no-CoT stay near chance (0.529/0.524) while explicit-CoT and implicit reach 0.892/0.900; on Chain, no-CoT (0.639) provides only a small gain over naive (0.612), compared to explicit-CoT/implicit (0.690/0.757); and on Collider, no-CoT is only modestly above naive (0.655 vs. 0.629) while explicit-CoT/implicit achieve 0.856/0.849. These results show that the intermediate causal reasoning steps is the main driver of improved performance on causal judgment questions. Meanwhile, the performance of **implicit causal reasoning** remains competitive and **surpass** explicit-CoT on confounder and chain dataset. While explicit-CoT and implicit causal reasoning occasionally produce invalid final-answer formats, our manual inspection shows all these invalid answers fail in matching any recognized formats of Yes or No (e.g., Yes., no!), or other identifiable decision keywords. In addition, the limited performance of no-CoT fine-tuning suggests that achieving success on CausalFlip requires the training strategy that effectively ground predictions in causal structure rather than exploiting spurious semantic correlations.

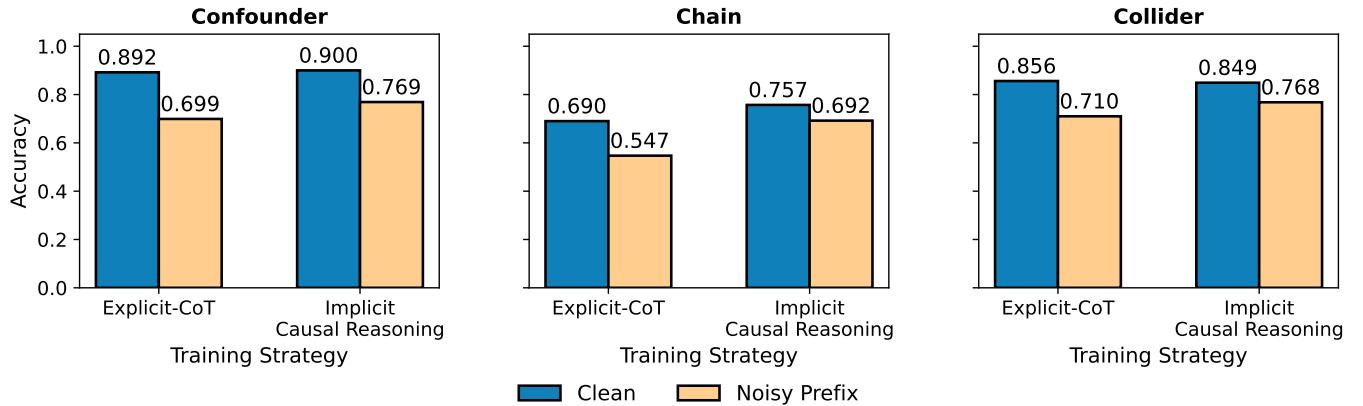
#### 5.5 Performance under Noisy Prefix (RQ2)

We next answer RQ2 by evaluating explicit-CoT and implicit causal reasoning’s performances on CausalFlip under the **noisy-prefix** injection (defined in section 5.3.4). This prefix is designed to be causally irrelevant that does not violate existing causal structure. Since the underlying causal structure is unchanged, the performance degradation indicates that the model’s prediction is sensitive to semantics, rather than being grounded in the causal structure.

As is shown in Figure 3, across all three datasets, adding the noisy prefix consistently causes a larger accuracy drop for explicit-CoT than for implicit causal reasoning, and the average drop is about 0.161 for explicit-CoT versus 0.092 for implicit causal reasoning. Moreover, under the noisy-prefix setting, implicit causal reasoning achieves higher accuracy than explicit-CoT on confounder, chain, and collider ( $0.769 > 0.699$ ,  $0.692 > 0.547$ ,  $0.768 > 0.710$ ). Overall, under the noisy semantic prefix setting, the larger degradation of explicit-CoT suggests that, it still predicts answers by relying heavily on semantics rather than the underlying causal structure. In contrast, implicit causal reasoning maintains higher accuracy under the same distraction, indicating stronger robustness in causality-related questions, and its reduced reliance on semantics.

## 6 Related Work

**Causal Inference in LLMs :** Recent work has investigated whether LLMs can perform causal inference [11, 15, 33]. In this line of research, the goal is to evaluate causal-effect judgments that depend on an underlying causal structure, for example, assessing whether a model can reason about causal links under different structural assumptions [10]. A common formulation is to pose causal questions in natural language while assuming a causal model behind the question [10, 15]. Prior work also explores how to elicit causal inference behavior from LLMs, including causal graphs and procedural guidance that implements parts of a causal inference process



**Figure 3: Accuracy of explicit-CoT versus implicit causal reasoning on CausalFlip across the three sub-datasets under clean inputs and noisy-prefix. Implicit causal reasoning consistently degrades less and perform better than explicit-CoT after the injection of noisy prefix, indicating its reduced reliance on spurious semantic correlations.**

[10, 15]. In this research context, our work complements prior studies on causal inference with LLMs in two ways. First, we introduce a training strategy that aims to reduce LLMs’ reliance on semantics in causality-related tasks. Second, we propose a benchmark for evaluating models’ causal reasoning ability: It formulates causal judgment questions over event triples, constructs semantically similar question pairs with opposite labels, and uses a pairwise train-test split so that semantic matching is systematically penalized.

**Causal Benchmarks for LLMs :** Prior work has proposed a range of benchmarks to evaluate causal understanding in LLMs: (i) *Commonsense cause-effect* benchmarks test whether models can select plausible cause or effect under everyday knowledge [25]. (ii) *Causality-text comprehension* benchmark tests whether models can apply causal knowledge from a passage to a new situation [18]. (iii) *Graph-grounded causal inference* benchmark evaluate model’s causal judgment using causal graphs and formal causal questions [10]. (iv) *Comprehensive causal reasoning* benchmark provides broader causal reasoning coverage across domains such as texts, math, etc., [33]. Finally, (v) *general LLM evaluation*, such as BIG-bench include causality-adjacent reasoning tasks and are often used to report general reasoning capabilities [28]. However, such benchmark accuracy can be an unreliable metric for true causal reasoning ability, due to the consistent finding that models can achieve high accuracy by exploiting spurious correlations with labels rather than reasoning from the underlying causal structure [17, 22, 31]. CausalFlip is designed to address this issue by using semantically similar, label-flipped pairs under a pairwise train-test split. Models that rely on spurious semantic correlations could only achieve limited performance; therefore, correct predictions need to be grounded in causal structure.

## 7 Conclusion

In this work, we address a key gap in evaluating LLM causal judgment, where standard autoregressive trained model can lean on label-correlated semantic patterns to answer causal questions, instead of grounding decisions in an underlying causal structure. To bridge this gap, we introduce **CausalFlip**, a causality-based

benchmark covering confounder, chain, and collider causal structures. Empirically, we find that strategies that heavily depend on semantic patterns fail on our benchmark, and whereas supervision on intermediate causal reasoning steps improves accuracy. More importantly, the proposed implicit causal reasoning strategy shows clear strengths by progressively masking intermediate causal reasoning step tokens from supervision to reduce model’s reliance on possible spurious semantic correlation in explicitly generated intermediate reasoning steps.

## 8 Ethics and Fairness

CausalFlip does not collect user data or involve human subjects, and it contains no private record or consent-required sources, such as identifiable information, medical or financial records, etc. During dataset construction, we manually inspected the data to mitigate risks of including sensitive information, offensive language, or potentially harmful stereotypes and biases in the text. We apply this review consistently throughout the full dataset to ensure that the released benchmark avoids such content to extent. Finally, this benchmark is intended for research evaluation purpose, and we encourage responsible use, including clear communication of limitations, and adherence to established ethical guidelines when developing downstream applications.

## References

- [1] Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Janvin. 2003. A neural probabilistic language model. *J. Mach. Learn. Res.* 3, null (March 2003), 1137–1155.
- [2] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language Models are Few-Shot Learners. arXiv:2005.14165 [cs.CL] <https://arxiv.org/abs/2005.14165>
- [3] Tianqi Chen, Bing Xu, Chiyuan Zhang, and Carlos Guestrin. 2016. Training Deep Nets with Sublinear Memory Cost. arXiv:1604.06174 [cs.LG] <https://arxiv.org/abs/1604.06174>
- [4] Jeffrey Cheng and Benjamin Van Durme. 2024. Compressed Chain of Thought: Efficient Reasoning Through Dense Representations. arXiv:2412.13171 [cs.CL]

- <https://arxiv.org/abs/2412.13171>
- [5] Yuntian Deng, Yejin Choi, and Stuart Shieber. 2024. From Explicit CoT to Implicit CoT: Learning to Internalize CoT Step by Step. arXiv:2405.14838 [cs.CL] <https://arxiv.org/abs/2405.14838>
- [6] Nouha Dziri, Ximing Lu, Melanie Sclar, Xiang Lorraine Li, Liwei Jiang, Bill Yuchen Lin, Peter West, Chandra Bhagavatula, Ronan Le Bras, Jena D. Hwang, Soumya Sanyal, Sean Welleck, Xiang Ren, Allyson Ettinger, Zaid Harchaoui, and Yejin Choi. 2023. Faith and Fate: Limits of Transformers on Compositionality. arXiv:2305.18654 [cs.CL] <https://arxiv.org/abs/2305.18654>
- [7] Angela Fan, Mike Lewis, and Yann Dauphin. 2018. Hierarchical Neural Story Generation. arXiv:1805.04833 [cs.CL] <https://arxiv.org/abs/1805.04833>
- [8] Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2020. The Curious Case of Neural Text Degeneration. arXiv:1904.09751 [cs.CL] <https://arxiv.org/abs/1904.09751>
- [9] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. LoRA: Low-Rank Adaptation of Large Language Models. arXiv:2106.09685 [cs.CL] <https://arxiv.org/abs/2106.09685>
- [10] Zhijing Jin, Yuen Chen, Felix Leeb, Luigi Gresele, Ojasv Kamal, Zhiheng Lyu, Kevin Blin, Fernando Gonzalez Adauto, Max Kleiman-Weiner, Mrinmaya Sachan, and Bernhard Schölkopf. 2024. CLadder: Assessing Causal Reasoning in Language Models. arXiv:2312.04350 [cs.CL] <https://arxiv.org/abs/2312.04350>
- [11] Zhijing Jin, Jiarui Liu, Zhiheng Lyu, Spencer Poff, Mrinmaya Sachan, Rada Mihalcea, Mona Diab, and Bernhard Schölkopf. 2024. Can Large Language Models Infer Causation from Correlation? arXiv:2306.05836 [cs.CL] <https://arxiv.org/abs/2306.05836>
- [12] Adam Tauman Kalai, Ofir Nachum, Santosh S. Vempala, and Edwin Zhang. 2025. Why Language Models Hallucinate. arXiv:2509.04664 [cs.CL] <https://arxiv.org/abs/2509.04664>
- [13] Dhairaj Kalamkar, Dheevatsa Mudigere, Naveen Mellempludi, et al. 2019. A Study of BFLOAT16 for Deep Learning Training. arXiv:1905.12322 [cs.LG] <https://arxiv.org/abs/1905.12322>
- [14] Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2023. Large Language Models are Zero-Shot Reasoners. arXiv:2205.11916 [cs.CL] <https://arxiv.org/abs/2205.11916>
- [15] Emre Kiciman, Robert Ness, Amit Sharma, and Chenhao Tan. 2024. Causal Reasoning and Large Language Models: Opening a New Frontier for Causality. arXiv:2305.00050 [cs.AI] <https://arxiv.org/abs/2305.00050>
- [16] Jinqi Lai, Wensheng Gan, Jiayang Wu, Zhenlian Qi, and Philip S. Yu. 2024. Large language models in law: A survey. *AI Open* 5 (2024), 181–196. doi:10.1016/j.aiopen.2024.09.002
- [17] Yuxuan Lai, Chen Zhang, Yansong Feng, Quzhe Huang, and Dongyan Zhao. 2021. Why Machine Reading Comprehension Models Learn Shortcuts? arXiv:2106.01024 [cs.CL] <https://arxiv.org/abs/2106.01024>
- [18] Kevin Lin, Oyvind Tafjord, Peter Clark, and Matt Gardner. 2019. Reasoning Over Paragraph Effects in Situations. arXiv:1908.05852 [cs.CL] <https://arxiv.org/abs/1908.05852>
- [19] Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. TruthfulQA: Measuring How Models Mimic Human Falsehoods. arXiv:2109.07958 [cs.CL] <https://arxiv.org/abs/2109.07958>
- [20] Ilya Loshchilov and Frank Hutter. 2019. Decoupled Weight Decay Regularization. arXiv:1711.05101 [cs.LG] <https://arxiv.org/abs/1711.05101>
- [21] Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. 2020. On Faithfulness and Factuality in Abstractive Summarization. arXiv:2005.00661 [cs.CL] <https://arxiv.org/abs/2005.00661>
- [22] R. Thomas McCoy, Ellie Pavlick, and Tal Linzen. 2019. Right for the Wrong Reasons: Diagnosing Syntactic Heuristics in Natural Language Inference. arXiv:1902.01007 [cs.CL] <https://arxiv.org/abs/1902.01007>
- [23] Daniel McDuff, Mike Schaeckermann, Tao Tu, Anil Palepu, Amy Wang, Jake Garrison, Karan Singhal, Yash Sharma, Shekoofeh Azizi, Kavita Kulkarni, Le Hou, Yong Cheng, Yun Liu, S Sara Mahdavi, Sushant Prakash, Anupam Pathak, Christopher Semturs, Shwetak Patel, Dale R Webster, Ewa Dominowska, Juraj Gottweis, Joelle Barral, Katherine Chou, Greg S Corrado, Yossi Matias, Jake Sunshine, Alan Karthikesalingam, and Vivek Natarajan. 2023. Towards Accurate Differential Diagnosis with Large Language Models. arXiv:2312.00164 [cs.CY] <https://arxiv.org/abs/2312.00164>
- [24] Alec Radford and Karthik Narasimhan. 2018. Improving Language Understanding by Generative Pre-Training. <https://api.semanticscholar.org/CorpusID:49313245>
- [25] Melissa Roemmele, Cosmin Adrian Bejan, and Andrew S. Gordon. 2011. Choice of Plausible Alternatives: An Evaluation of Commonsense Causal Reasoning. In *AAAI Spring Symposium on Logical Formalizations of Commonsense Reasoning*. Stanford University. <http://ict.usc.edu/pubs/Choice%20of%20Plausible%20Alternatives%20An%20Evaluation%20of%20Commonsense%20Causal%20Reasoning.pdf>
- [26] Junhan Shi, Yijia Zhu, Zhenning Shi, Dan Zhao, Qing Li, and Yong Jiang. 2025. SpecCoT: Accelerating Chain-of-Thought Reasoning through Speculative Exploration. In *Findings of the Association for Computational Linguistics: EMNLP 2025*. Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (Eds.). Association for Computational Linguistics, Suzhou, China, 24405–24415. doi:10.18653/v1/2025.findings-emnlp.1326
- [27] Karan Singhal, Shekoofeh Azizi, Tao Tu, S. Sara Mahdavi, Jason Wei, Hyung Won Chung, Nathan Scales, Ajay Tanwani, Heather Cole-Lewis, Stephen Pfohl, Perry Payne, Martin Seneviratne, Paul Gamble, Chris Kelly, Nathaneal Scharli, Aakanksa Chowdhery, Philip Mansfield, Blaise Aguera y Arcas, Dale Webster, Greg S. Corrado, Yossi Matias, Katherine Chou, Juraj Gottweis, Nenad Tomasev, Yun Liu, Alvin Rajkomar, Joelle Barral, Christopher Semturs, Alan Karthikesalingam, and Vivek Natarajan. 2022. Large Language Models Encode Clinical Knowledge. arXiv:2212.13138 [cs.CL] <https://arxiv.org/abs/2212.13138>
- [28] Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, et al. 2023. Beyond the Imitation Game: Quantifying and extrapolating the capabilities of language models. arXiv:2206.04615 [cs.CL] <https://arxiv.org/abs/2206.04615>
- [29] Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to Sequence Learning with Neural Networks. arXiv:1409.3215 [cs.CL] <https://arxiv.org/abs/1409.3215>
- [30] Mirac Suzgun, Nathan Scales, Nathanael Schärlí, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksa Chowdhery, Quoc V. Le, Ed H. Chi, Denny Zhou, and Jason Wei. 2022. Challenging BIG-Bench Tasks and Whether Chain-of-Thought Can Solve Them. arXiv:2210.09261 [cs.CL] <https://arxiv.org/abs/2210.09261>
- [31] Ruixiang Tang, Dehan Kong, Longtao Huang, and Hui Xue. 2023. Large Language Models Can be Lazy Learners: Analyze Shortcuts in In-Context Learning. In *Findings of the Association for Computational Linguistics: ACL 2023*. Association for Computational Linguistics, 4645–4657. doi:10.18653/v1/2023.findings-acl.284
- [32] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2023. Attention Is All You Need. arXiv:1706.03762 [cs.CL] <https://arxiv.org/abs/1706.03762>
- [33] Zeyu Wang. 2024. CausalBench: A Comprehensive Benchmark for Evaluating Causal Reasoning Capabilities of Large Language Models. In *Proceedings of the 10th SIGHAN Workshop on Chinese Language Processing (SIGHAN-10)*, Kam-Fai Wong, Min Zhang, Ruifeng Xu, Jing Li, Zhongyu Wei, Lin Gui, Bin Liang, and Runcong Zhao (Eds.). Association for Computational Linguistics, Bangkok, Thailand, 143–151. <https://aclanthology.org/2024.sighan-1.17/>
- [34] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2023. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. arXiv:2201.11903 [cs.CL] <https://arxiv.org/abs/2201.11903>
- [35] Shijie Wu, Ozan Irsoy, Steven Lu, Vadim Dabrowski, Mark Dredze, Sebastian Gehrmann, Praphanjan Kambadur, David Rosenberg, and Gideon Mann. 2023. BloombergGPT: A Large Language Model for Finance. arXiv:2303.17564 [cs.LG] <https://arxiv.org/abs/2303.17564>
- [36] Hongyang Yang, Xiao-Yang Liu, and Christina Dan Wang. 2025. FinGPT: Open-Source Financial Large Language Models. arXiv:2306.06031 [q-fin.ST] <https://arxiv.org/abs/2306.06031>
- [37] Dun Zhang, Jiacheng Li, Ziyang Zeng, and Fulong Wang. 2025. Jasper and Stella distillation of SOTA embedding models. arXiv:2412.19048 [cs.IR] <https://arxiv.org/abs/2412.19048>

## A Appendix

### A.1 Reducing Data Skewness

Although the pairwise causal questions with opposite causal answers are specifically designed to be semantically close to each other, the dataset as a whole can still contain skewed correlations between semantic features and labels that LLMs could learn as spurious shortcuts [17, 31]. Here, “skewness” refers to distributional imbalances where particular event appear disproportionately in certain categories. For example, certain event can still show up more in one category than the others, such as sales of certain products or certain weather conditions appearing more as event in Base pairs than Opposite pairs. Such imbalances can create shortcut signals that allow models to predict labels without grounding the reasoning in causal structure.

We detect such skewness from two perspectives. First, we analyze the *count* of event in each dataset, computing their frequencies across labels and categories and flagging those with highly imbalanced distributions under a threshold. In addition, we perform *similarity-based* analysis to identify examples that dominate the

dataset’s semantic space. Specifically, we embed each question using the `stella-400M` model [37] and compute cosine similarity; for each example, we retrieve its top-5 nearest neighbors in the dataset. We then count how often an example appears in the top-5 neighbor lists of other examples. Examples that appear unusually frequently in these top-5 lists indicate near-duplicate clusters or highly generic patterns that may amplify label-correlated shortcuts.

To mitigate these skewness, we filter and replace the top 5 cases identified by the two analyses iteratively. For count-based skew, we replace overrepresented event (in  $X$ ,  $Y$ , or  $Z$ ) with substitute events so that event frequencies are more evenly distributed across categories. For similarity-based analysis, we replace examples that

appear unusually often as a top-5 neighbor of many others, reducing skewness induced by similarity and preventing highly repetitive patterns from dominating the training shortcuts.

## A.2 Hyperparameters

For all fine-tuned strategies (no-CoT, explicit-CoT and implicit causal reasoning), we apply Low-Rank Adaptation (LoRA) [9] with  $r = 4$ ,  $\alpha = 8$ ,  $\text{dropout} = 0.05$ , targeting  $q\_\text{proj}$ ,  $v\_\text{proj}$ . We fine-tune for 3 epochs with learning rate  $1 \times 10^{-4}$ , batch size 4, maximum sequence length 256, using the paged AdamW [20] 8-bit optimizer, bf16 training [13], and gradient checkpointing [3], with fixed random seed for fair comparisons.