How Likely Do LLMs with CoT Mimic Human Reasoning?

Guangsheng Bao[1,2,*], **Hongbo Zhang**[1,2,*], **Cunxiang Wang**[1,2], **Linyi Yang**[2,3], **Yue Zhang**[2,3,†]
[1] Zhejiang University
[2] School of Engineering, Westlake University
[3] Institute of Advanced Technology, Westlake Institute for Advanced Study
{baoguangsheng,zhanghongbo,wangcunxiang,yanglinyi,zhangyue}@westlake.edu.cn

Abstract

Chain-of-thought emerges as a promising technique for eliciting reasoning capabilities from Large Language Models (LLMs). However, it does not always improve task performance or accurately represent reasoning processes, leaving unresolved questions about its usage. In this paper, we diagnose the underlying mechanism by comparing the reasoning process of LLMs with humans, using causal analysis to understand the relationships between the problem instruction, reasoning, and the answer in LLMs. Our empirical study reveals that LLMs often deviate from the ideal causal chain, resulting in spurious correlations and potential consistency errors (inconsistent reasoning and answers). We also examine various factors influencing the causal structure, finding that in-context learning with examples strengthens it, while post-training techniques like supervised fine-tuning and reinforcement learning on human feedback weaken it. To our surprise, the causal structure cannot be strengthened by enlarging the model size only, urging research on new techniques. We hope that this preliminary study will shed light on understanding and improving the reasoning process in LLM. [1]

1 Introduction

Chain of thought (CoT) has become a standard technique for using LLM to solve reasoning tasks (Wei et al., 2022b; Kojima et al., 2022; Wang et al., 2023), including complex mathematical reasoning (Cobbe et al., 2021; Lewkowycz et al., 2022; Imani et al., 2023) and logical reasoning (Liu et al., 2023b; Xu et al., 2023; Pan et al., 2023). However, studies show that CoT does not uniformly lead to increased performance (Kojima et al., 2022; Sprague et al., 2023) and does not always faithfully represent the true reasoning process in LLM (Lanham et al., 2023; Turpin et al., 2023), leaving unsolved questions such as *when* and *why* if these issues occur. Intuitively, understanding the mechanism behind and identifying the root cause will be useful in fixing the issues.

Part	Problem Solving Process
Instruction (Z)	Please act as a math teacher and solve the addition problem ... # Question: What is the sum of 625126 and 542611? # Reasoning:
CoT (X)	Let's add the two numbers digit by digit. 1. The ones place: 6 + 1 = 7 2. The tens place: 2 + 1 = 3 ...
Answer (Y)	Therefore, the final computed sum is 1167737.

Table 1: A simple example for the three parts of a general problem-solving process.

While existing studies have focused on reasoning at the phenomena level (Jin et al., 2024; Harsha Tanneru et al., 2024; Yee et al., 2024), we employ a causal approach, revealing the underlying mechanism of CoT and comparing it with human reasoning. Specifically, without losing generality between tasks and questions, we abstract problem solving into three parts: problem *instruction*, reasoning steps (*CoT*), and conclusion (*answer*), with each part denoted by a random variable that Z for instruction, X for CoT, and Y for answer, as a simple example shown in Table 1. We discuss the causal relationship between the three variables for both humans and LLMs. Studies suggest that rational humans follow a *causal chain* when solving complex reasoning problems (Cummins, 1995; Hegarty, 2004; Sloman and Lagnado, 2015), where the instruction causes the reasoning steps and the reasoning steps cause the conclusion.

For LLMs, we perform *causal analysis* against the three variables by employing interventions (Hagmayer et al., 2007), assessing the significance

[*] Equal contribution. [†] Corresponding author.
[1] We release the code and results at https://github.com/StevenZHB/CoT_Causal_Analysis

Proceedings of the 31st International Conference on Computational Linguistics, pages 7831–7850
January 19–24, 2025.

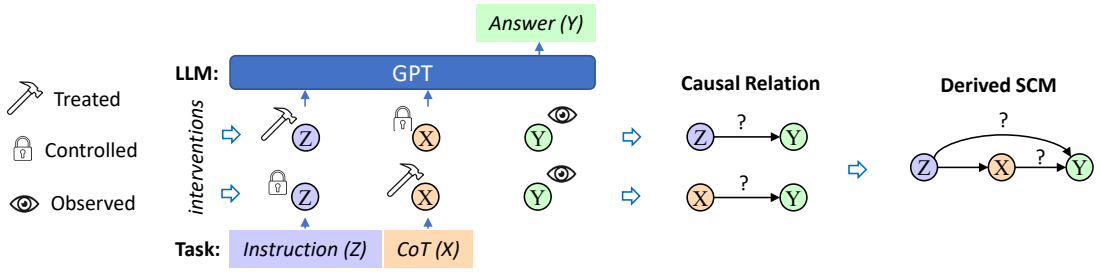

Figure 1: *Causal analysis*, where we identify an SCM from an LLM-task pair using treatment experiments. For each pair of variables with possible causal relation, we conduct an experiment by injecting an intervention into the treated variable and observe its effect.

of the cause-effect relationship between each pair of variables and assembling a structural causal model (SCM) (Pearl, 2009) for each LLM and task pair, as illustrated in Figure 1. Specifically, we reveal four types of SCM, including causal chain, common cause, full connection, and isolation (Figure 2). Experiments show that a significant portion of LLM-task pairs have the types of common cause (II) and full connection (III), where the model suffers from unexpected spurious correlations between instruction and answer (as the arcs from Z to Y in the SCMs). Empirical evidence indicates that LLMs in these cases may not actually do reasoning (conclude the answer from the CoT) but do explaining (produce the CoT according to the latent belief of the answer). Therefore, reasoning processes can cause an inconsistency error (mismatch between CoT and the answer) and an unfaithful response (discrepancy between CoT and the true reason), as mentioned in the column '*consistency & faithfulness*' in Figure 2.

We further investigate various factors that possibly influence the causal structure of implied SCM in six tasks, finding that in-context learning (ICL) strengthens the causal structure while supervised fine-tuning (SFT) (Wei et al., 2022a) and reinforcement learning on human feedback (RLHF) (Ouyang et al., 2022) weakens it. In addition, our investigation of different model sizes reveals that larger language models may not imply stronger type of SCM, suggesting that enlarging model size only may not lead LLMs to ideal human-level reasoning abilities.

Our contributions are mainly twofold:

1) We discovered *the underlying SCMs of LLMs as essential features*, forecasting their superficial behaviors, such as making consistency errors and producing unfaithful explanations.

2) We investigated relevant factors suggesting that *human-level reasoning ability may not be reached by enlarging the model size of LLMs* and popular post-training techniques such as SFT and RLHF actually weaken it.

2 Related Work

LLM Reasoning. Various reasoning techniques have been proposed to enhance the reasoning ability of LLMs (Chu et al., 2023; Yu et al., 2024). Chain-of-thought (CoT) prompting (Wei et al., 2022b), as an early study elicits reasoning in LLMs, inspires numerous further investigations. Specifically, self-consistency (Wang et al., 2022) votes the major decision from multiple reasoning paths, Tree-of-thought (Yao et al., 2023a) searches the most confident reasoning path in a tree, and Graph-of-thought (Yao et al., 2023b) represents the thoughts as graph nodes and combines thoughts non-sequentially. Advanced CoT methods, like Faithful CoT (Lyu et al., 2023) and Constraint CoT (Vacareanu et al., 2024), are further proposed to improve reasoning capabilities. In this paper, we focus on the very basic chain of thought to understand the underlying mechanism of how LLMs do reasoning, leaving the analysis of advanced methods for the future.

Chain-of-Thought Faithfulness. Various recent studies raise concerns about the faithfulness of the reasoning steps generated by CoT prompting. Among them, Turpin et al. (2023) elicits unfaithful reasoning steps using biased model inputs, Paul et al. (2024) finds that the generated answer may not rely on the reasoning steps, Jin et al. (2024) lengthens the reasoning steps without adding new information but improves the accuracy of the task, Pfau et al. (2024) replaces the reasoning steps with filler tokens to solve algorithmic tasks. Furthermore, Lanham et al. (2023) proposes a metric to

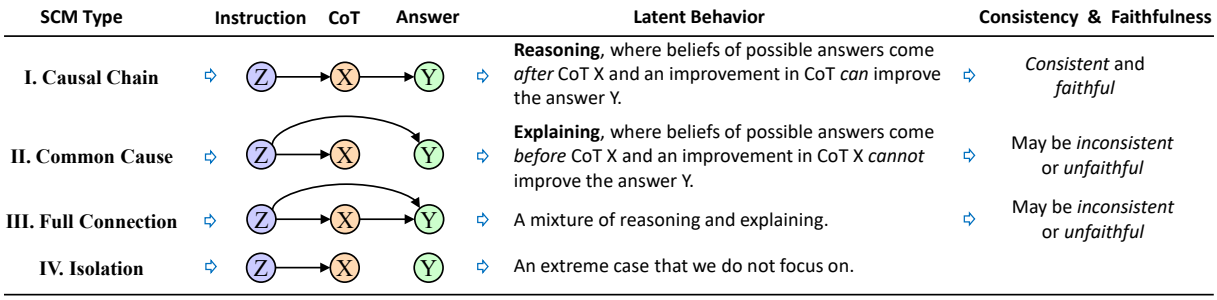

Figure 2: *Four types of SCM*, where the structure of an SCM reveals its latent behavior, providing explanations on *when* and *why* problems may occur during the reasoning process.

measure the faithfulness of CoT reasoning by intervening the CoT with injected mistakes and altered expressions, but lately Bentham et al. (2024) doubts about the validity of the metric due to its huge variation under small changes. These studies identify the unfaithfulness of CoT or try to measure it. In this paper, we go further to analyze the latent SCM structures from which we draw connections to various effects, including consistency, faithfulness, and task accuracy.

Causal Reasoning in LLMs. Existing studies on the causal reasoning capabilities of LLM mainly focus on variables described in natural language, like benchmark ATOMIC (Sap et al., 2019), CLadder (Jin et al., 2023a), and Corr2Cause (Jin et al., 2023b). ATOMIC focuses on the if-then relations of variables like "if X pays Y, Y will probably return". CLadder requires the identification of variables and their causal relations from the language context prior to inference. Corr2Cause determines the causal structure between variables according to a group of correlational statements. On multiple causal benchmarks, Kıcıman et al. (2023) finds that LLMs achieve good accuracy and hypothesizes that LLMs can use their collected knowledge to generate causal graphs from natural language, while Zečević et al. (2023) conjectures that a successful causal inference relies on a pre-learned meta-SCM that stores the related causal facts in natural language. Unlike these studies, where variables represent targets in the question domain, we investigate the causality between three known variables, instruction, CoT, and answer, which do not represent any question-specific target, but only abstractive components of the chain-of-thought reasoning.

3 Causal Analysis

The causal analysis involves three random variables, two hypotheses, and four types of SCM. We explain relevant terminologies such as variables, spurious correlation, and causal relationship in Appendix A and the basic ideas of causal analysis in Appendix B, including the definition of SCM and confounder.

3.1 Random Variables

As demonstrated in Table 1, the problem solving process can be broken down into three random variables, assuming that the question, task, and model remain constant during each experiment.

Instruction (Z) typically includes a task description, a few demonstrations, and a question formulation, which guides LLMs in generating a solution or a response. The instruction is restricted by task and question, but the description, demonstrations, and expression can be altered in each experiment.

CoT (X) signifies the step-by-step reasoning process of an LLM, which results in an answer that is generally considered more precise than a direct answer produced by the LLM. To examine this belief, we distinguish the CoT and the answer as two variables in this paper.

Answer (Y) symbolizes the final step of the reasoning process, which answers the question. Ideally, the answer is fully determined by the reasoning steps, which provide complete evidence for the final decision. The answer step is different for each task. For example, "*The correct option is: A*" for multiple choice tasks, "*The answer is 10.*" for GSM8K and "*Therefore, the final computed sum is 100.*" for Addition. Some complete examples are shown in Figure 3 in the Appendix.

3.2 Identification of SCM

Intuitively, an autoregressive language model enables the right tokens to depend on all the left tokens, which are represented as a full connection. However, for each specific task, a language model could potentially work in any subgraph of the full

SCM. To imply the underlying SCM type of an LLM in a task, we test the causal relations using interventions, focusing on the answer and the relations pointing to it, as illustrated in Figure 1.

Definition 3.1 (Cause-Effect Interventions) *Suppose that the SCM $\mathcal{G}$ entails a distribution $P_{X,Y}$ with $N_X, N_Y \overset{iid}{\sim} \mathcal{N}(0,1)$. Then we intervene on X to change the distribution of Y*

$$P_Y^{do(X)} = P(Y|do(X)). \tag{1}$$

Using interventions independent of other variables, we decide whether the treated variable X causes the target variable Y.

Definition 3.2 (Average Treatment Effect) *An ATE (Rubin, 1974) represents the effect of an intervention, which compares the distributions of the target variable Y with and without a treatment.*

$$ATE = E(Y|do(X)) - E(Y). \tag{2}$$

We assess the significance of the average treatment effects (Angrist and Imbens, 1995) using McNemar's test (McNemar, 1947). Specifically, we test two hypotheses: *if the CoT in LLMs causes the answer* and *if the instruction causes the answer*.

Hypothesis 3.1 (CoT causes Answer) *Given a constant Instruction,*

$$\begin{cases} H_0 : ATE = 0, & \textit{CoT does not cause Answer,} \\ H_1 : ATE \neq 0, & \textit{CoT causes Answer,} \end{cases} \tag{3}$$

where $ATE = E(Y|Z, do(X)) - E(Y|Z)$.

Hypothesis 3.2 (Instruction causes Answer) *Given a constant CoT,*

$$\begin{cases} H_0 : ATE = 0, & \textit{Instruction does not cause Answer,} \\ H_1 : ATE \neq 0, & \textit{Instruction causes Answer,} \end{cases} \tag{4}$$

where $ATE = E(Y|X, do(Z)) - E(Y|X)$.

Based on the statistical significance of the hypotheses, we infer the underlying SCM for LLMs in each task.

Specifically, we test hypothesis 3.1 using two contrasting interventions, golden CoT and random CoT, with expected effects of improving and altering task accuracies, respectively. The **golden CoT** represents the reasoning steps with golden quality, from which we test if an LLM can conclude the answer correctly. The golden CoT is provided by only four of the datasets. The **random CoT** is created based on the LLM-generated CoT by injecting mistakes, inspired by (Lanham et al., 2023). We randomly replace the numbers in mathematical problems with new digits, and turn the last 1/3 of assertions in logical reasoning problems into negative expressions. More design considerations for CoT interventions are discussed in Appendix B.1.

Furthermore, we test hypothesis 3.2 under two conditions, the default CoT and the golden CoT, where the CoTs remain constant during the treatment experiments. The **default CoT** refers to the CoT generated by the LLM using the default instruction. For each condition, we intervene in the instructions either through random instruction or random bias. The **random instruction** is different for each task, which is generated by GPT-4 based on the default instruction using a predefined paraphrasing prompt, inspired by the CoT paraphrasing (Lanham et al., 2023). We generate a list of alternative instructions for each task and randomly select one for each sample. The design considerations are described in Appendix B.2. The **random bias** is designed to lead LLMs to wrong answers. We follow (Turpin et al., 2023) to inject a bias statement in the instruction like "*I think the correct answer/option is: <the answer>*", where the answer is either a corrupted golden answer (a digit in the answer is randomly replaced) or a wrong option (a random choice different from the golden selection).

3.3 SCM Types

As shown in Figure 2, the three random variables can potentially be connected in four directional acyclic graphs (DAGs), each representing a type of SCM.

Type I. Causal Chain is the ideal SCM in which we anticipate LLMs to operate, where the answer Y is entirely determined by CoT X. The instruction Z determines the CoT but NOT the answer directly. In other words, the reasoning steps (CoT) fully determine the answer given a question, while other auxiliary information, such as task descriptions and demonstrations, do not directly influence the answer. In such a deterministic relation, we say that CoT *causes* the answer, and CoT is the *only cause* of the answer.

Type II. Common Cause represents an SCM with a confounder instruction Z, where the CoT X and the answer Y are not causally connected. We say that the CoT and the answer are conditionally independent given the instruction. Such disconnected CoT and answer are hard to identify from the response only until they produce observable consistency errors.

Type III. Full Connection is the SCM that an autoregressive LLM generally mimics, where the left-to-right causal direction is supported by the causal attention mask (Vaswani et al., 2017) of LLMs. Basically, an LLM is capable of simulating any subgraph of the full SCM. However, the statistical learning of pretraining may not catch the underlying causal structure behind the observation, resulting in superficial behavior-level simulation of human step-by-step reasoning.

Type IV. Isolation denotes an extreme case of SCM, where the answer Y is not influenced by either the instruction Z or CoT X. This means that the answer is either determined by the question directly or generated randomly. Due to the complexity of this case, we leave it out of our focus.

4 Experiments

4.1 Experimental Settings

We evaluate the performance of the model in tasks in terms of *accuracy* by comparing the generated responses with the reference responses. See Appendix C.1 for a detailed description of the prompts used.

Models. We select multiple open-source and API-based models in experiments, including *ChatGPT* (OpenAI, 2022), *GPT-4* (OpenAI, 2023)[2], and open-source models such as the *Llama2* family (Touvron et al., 2023) and the *Mistral* series (Jiang et al., 2023). For proprietary models, we use official API calls[3]. For open-source models, we employ vLLM (Kwon et al., 2023) for local deployment. We use a temperature of 0.0 for all experiments.

Datasets. We use six reasoning tasks, primarily mathematical and logical, which are expected to have straightforward, unbiased, and organized solutions. The mathematical tasks consisted of basic arithmetic calculations and math word problems, where LLMs demonstrate shortcomings in the former (Qian et al., 2022) but excel in the latter (Wei et al., 2022b). We create groups of n-digit numbers for elementary-level arithmetic calculations, specifically *Addition* and *Multiplication*. For each of the 6-digit and 9-digit additions and the 2-digit and 3-digit multiplications, 500 samples are generated. Each sample includes a golden step-by-step calculation. For math word problems, we randomly select 500 samples from the *GSM8K* dataset (Cobbe et al., 2021).

We utilize three datasets for the logical tasks. *ProofWriter* (Tafjord et al., 2020) is commonly used for deductive logical reasoning. We randomly select 600 instances from the 5-hop reasoning development set. *FOLIO* (Han et al., 2022) is another dataset for deductive logical reasoning, notable for its expertly crafted content that is more reflective of real-world scenarios. We use all 204 instances from the development set. *LogiQA* (Liu et al., 2023a) is a dataset drawn from questions in the verbal reasoning exam that require complex logical reasoning abilities. We randomly select 600 entries from the LogiQA 2.0 test set.

Evaluation of Consistency Error. We evaluate the consistency between the CoTs and answers. For arithmetic tasks, reasoning steps (CoTs) are extracted and converted into equations using GPT-3.5-turbo prompts (see Appendix C.2). The generated equations are then compared to the standard (golden) equations to determine the correctness of the generated CoTs. We use the full datasets and report the ratio for each type of error.

For GSM8K and logical tasks, we perform a manual evaluation. A CoT is considered incorrect if it exhibits logical fallacies, contains factual inaccuracies, or fails to deduce the correct answer from the question. We randomly select 200 instances per task and manually examine the generated CoTs by two independent checkers (authors), achieving 96% agreement on 100 overlapped instances.

4.2 Causal Structures in LLM Tasks

Deriving SCM from treatment experiments. Taking *GPT-3.5-turbo* as an example, we illustrate how to interpret the treatment results. As shown in Table 2, by testing the two hypotheses, we obtain qualitative indicators (with a value of '*T*' for true or '*F*' for false) of causal relationships in each task. Based on these indicators, we imply the SCM. Specifically, the GPT-3.5-turbo in GSM8K and FOLIO exhibits type I SCM, where answers are NOT significantly affected by interventions in instructions but are significantly affected by interventions in CoTs, suggesting that the LLM in these tasks tends to do real reasoning.

GPT-3.5-turbo in Addition task implies type II SCM, where the answers depend on the instructions but not on the CoTs. Treatment with golden CoTs does not improve accuracy and treatment

[2] gpt-3.5-turbo-0613 and gpt-4-0613, respectively.
[3] https://openai.com/blog/openai-api

Experiment	GPT-3.5-Turbo						
	Add.	**Mult.**	**GSM8K**	**ProofWriter**	**FOLIO**	**LogicQA**	**Avg. |ATE|**
Zero-Shot CoT (baseline)	0.674	0.450	0.748	0.518	0.574	0.465	-
Hypothesis Test: If the CoT causes the answer given a constant instruction?							
Controlled (w/ default setting)	0.782	0.454	0.742	0.520	0.588	0.480	-
Treated (w/ golden CoT)	*0.768*	*0.638*	*1.000*	*0.777*	-	-	-
	(-0.014)	*(+0.184)**	*(+0.258)**	*(+0.257)**	-	-	*(0.178)*
Treated (w/ random CoT)	*0.764*	*0.000*	*0.018*	*0.427*	*0.495*	*0.440*	-
	(-0.018)	*(-0.454)**	*(-0.724)**	*(-0.093)**	*(-0.093)**	*(-0.040)**	*(0.237)*
CoT $\xrightarrow{?}$ Answer	**F**	**T**	**T**	**T**	**T**	**T**	(0.208)
Hypothesis Test: If the instruction causes the answer given a constant CoT?							
Controlled (w/ default CoT)	0.782	0.454	0.742	0.520	0.588	0.480	-
Treated (w/ random instruction)	*0.532*	*0.488*	*0.742*	*0.517*	*0.578*	*0.473*	-
	*(-0.250)**	*(+0.034)*	*(+0.000)*	*(-0.003)*	*(-0.010)*	*(-0.007)*	*(0.051)*
Treated (w/ random bias)	*0.228*	*0.412*	*0.746*	*0.503*	*0.563*	*0.433*	-
	*(-0.554)**	*(-0.042)*	*(+0.004)*	*(-0.017)**	*(-0.025)*	*(-0.047)**	*(0.115)*
Controlled (w/ golden CoT)	0.768	0.642	1.000	0.782	-	-	-
Treated (w/ random instruction)	*0.510*	*0.648*	*1.000*	*0.612*	-	-	-
	*(-0.258)**	*(+0.006)*	*(+0.000)*	*(-0.170)**	-	-	*(0.109)*
Treated (w/ random bias)	*0.198*	*0.488*	*1.000*	*0.650*	-	-	-
	*(-0.570)**	*(-0.154)**	*(+0.000)*	*(-0.132)**	-	-	*(0.214)*
Instruction $\xrightarrow{?}$ Answer	**T**	**T**	**F**	**T**	**F**	**T**	(0.122)
Implied SCM Type	**II**	**III**	**I**	**III**	**I**	**III**	-

Table 2: *Identification of causal structures* in tasks running on *GPT-3.5-Turbo*, where we present task accuracy and ATE. We test the significance of the ATEs from the treated experiments, where the asterisk '*' denotes a statistically significance with a p-value < 0.01 in McNemar's test. The term '*default setting*' denotes the default instruction predefined and the default CoT produced by the LLM. It is worth noting that the classification of SCM types is not strictly defined but in the sense of statistical significance.

with random CoTs does not reduce accuracy, suggesting the independence between CoT and answer. In this case, the CoT actually performs explaining instead of reasoning. GPT-3.5-turbo in Multiplication, ProofWriter, and LogiQA implies the full connection, where the latent behavior of CoT is a mixture of explaining and reasoning.

Distribution of SCM types. We further collect SCM types for other LLMs such as *GPT-4*, *Llama2-7B-Chat* and *Llama2-70B-Chat*, where the experiments are presented in Table 10, 11, and 12, respectively, in Appendix F. From these experiments, we obtain the distribution of SCM types.

As shown in Table 3, different LLMs suggest different types of SCM. Among them, type III (full connection) is the most common case (10 out of 24 LLM tasks), indicating that most LLMs perform a mixed behavior of reasoning and explaining. In smaller Llama2 models, the inferred SCMs are more likely to be type II, III, and IV rather than type I. Although larger GPT-3.5-turbo and GPT-4 show more times of type I, they still have a significant portion in types II and IV. Consequently, larger LLMs do not necessarily approach the ideal causal chain, suggesting that enlarging the model size only may not lead LLMs to human-level reasoning.

LLMs have different SCM types in different tasks, suggesting their inconsistent ability in different tasks. In this sense, the estimated SCM provides a meaningful indicator of the ability of the model, which can predict the possible mistakes that the model may produce.

SCM	Meta Llama2		OpenAI GPT		#LLM-Tasks
	Chat/7B	**Chat/70B**	**GPT3.5/175B**	**GPT4/?B**	
I	-	-	GSM8K FOLIO	GSM8K Mult.	4
II	Addition FOLIO LogiQA	-	Addition	Addition	5
III	GSM8K ProofW.	GSM8K ProofW. Addition Mult.	LogiQA ProofW. Mult.	ProofW.	10
IV	Mult.	FOLIO LogiQA	-	FOLIO LogiQA	5

Table 3: *Distribution of SCM types*, where the larger models do not necessarily imply stronger SCM types.

4.3 When and why do the issues happen?

We argue that the SCM is an essential feature of an LLM task pair, revealing latent behavior and predicting various superficial problems.

Link to Task Performance. Interestingly, the task accuracy of an LLM is not directly related to the type of SCM. When we compare GPT-4 with GPT-3.5-turbo, although GPT-4 achieves a relatively 41% higher average task accuracy (Table 8 in Appendix D), its inferred SCMs do not exhibit more in type I. The type of SCM determines the reasoning process, but not the task accuracy directly.

Consequently, we need different strategies to improve the accuracy of the answers for different types of SCM. For SCM type I, it can be achieved by enhancing the quality of the reasoning steps. However, for SCM type II, because of the conditional independence between the CoT and the answer, it is impossible to achieve better task accuracy by improving the CoT. These analyses are supported by the treatment experiments with golden CoT, as GSM8K (type I) and Addition (type II) in Table 2 shows. The golden CoTs (treated w/ golden CoT) improves the task accuracy of GSM8K from 0.742 to 1.000 (+0.258), but does not improve the task accuracy of Addition.

For SCM type III, it is also possible to improve accuracy by improving the reasoning steps, but there is no guarantee due to the unknown portion of reasoning and explaining underlying the CoT. Specifically, as the Multiplication and ProofWriter in the table show, the treatment with golden CoT improves the task accuracy of Multiplication from 0.454 to 0.638 (+0.184) and ProofWriter from 0.520 to 0.777 (+0.257), where improvements are made, but accuracies are still far from perfect 1.

Link to Faithfulness. Given an SCM type, we can predict the faithfulness of the LLM responses. For type I, the LLM tends to produce faithful reasons, while for type II and III, the LLM may produce unfaithful explanations because of the confounder between the CoT and the answer. These forecasts are confirmed by the significant ATEs under random bias treatment, as shown on Addition, Multiplication, ProofWriter, and LogiQA in Table 2. The bias changes the beliefs of the answer before CoTs and the answers. As a result, even with constant CoTs (either the default CoT or golden CoT), the answers change into incorrect ones, demonstrating an unfaithful representation of CoT to the real reasoning behind the latent belief.

In practice, none of the LLMs and tasks performs pure reasoning or explaining, but is somehow a mixture of them (as supported by the insignificant but non-zero ATE values in Table 2). Therefore, unfaithful responses generally occur in all LLMs and tasks to some extent.

Link to Consistency Error. We evaluate the consistency of CoTs and answers in six tasks, finding that incorrect CoTs may be followed by correct answers and vice versa. As Table 9 in the Appendix shows, on five of the six tasks, LLMs produce consistency errors, particularly on simple arithmetic problems like Addition and Multiplication. For example, more than 60% incorrect CoTs lead to correct answers in Addition, and a larger model such as GPT-4 shows even more discrepancy of 74% (Appendix D.2).

SCM	Behavior	GPT-3.5-Turbo		GPT-4	
		Task	Error Rate	Task	Error Rate
I	Reasoning	GSM8K	0.000	GSM8K	0.000
		FOLIO	0.125	Multi	0.178
II	Explaining	Addition	0.648	Addition	0.744
III	Mixture	LogiQA	0.125		
		ProofWriter	0.280	ProofWriter	0.060
		Multi	0.444		

Table 4: *Correspondence between SCM type and consistency error*, where the intensity of red denotes the severity of the consistency errors, with darker colors indicating higher rates. Overall, the reasoning behavior corresponds to the least error rate, while the explaining behavior the most and the mixture behavior the middle.

Intuitively, the reasoning behavior tends to produce consistent responses because the answers are concluded from the reasoning steps, while the explaining behavior may produce inconsistent CoTs and answers because they stochastically depend on the same latent beliefs. We examine the types of SCM and the consistency error rates as shown in Table 4. The results indicate that the tasks with type I SCM generally have lower error rates than the tasks with type II SCM, suggesting that the former makes fewer consistency errors because of the strong causal connection between the CoT and the answer. The untypical SCM type III is expected to sit between types I and II because of the mixed behavior, which is partially supported by the error rates (larger than or equal to type I generally but smaller than type II).

4.4 How to fix the issues?

We investigate possible factors that influence causal structures, including ICL, SFT, and RLHF.

Impact of In-Context Learning. In-context learning (ICL) is commonly employed to elicit expected behaviors in LLM (Brown et al., 2020), which is also used to trigger CoT to mimic human step-by-step reasoning (Wei et al., 2022b). We assess the impact of ICL on causal relationships in our context, with randomly chosen examples as ICL demonstrations. We carry out treatment experiments with varying numbers of demonstrations, as shown in Table 5.

ICL	|ATE| on 'CoT $\xrightarrow{?}$ Answer' (↑)							|ATE| on 'Instruction $\xrightarrow{?}$ Answer' (↓)							Task
	Add.	Mul.	GSM.	Pro.	FOL.	LQA.	Avg.	Add.	Mul.	GSM.	Pro.	FOL.	LQA.	Avg.	Accuracy
0-shot	0.016_F	0.319_T	0.491_T	0.175_T	0.093_T	0.040_T	0.208	0.408_T	0.059_T	0.001_F	$\mathbf{0.081}_T$	$\mathbf{0.018}_F$	0.027_T	0.122	0.572
2-shot	$\mathbf{0.095}_T$	0.338_T	0.497_T	0.254_T	$\mathbf{0.265}_T$	$\mathbf{0.043}_T$	**0.251**	0.104_T	0.035_T	$\mathbf{0.000}_F$	0.117_T	0.073_T	0.006_F	0.059	**0.598**
4-shot	0.014_F	0.336_T	$\mathbf{0.498}_T$	0.251_T	0.235_T	0.022_F	0.227	$\mathbf{0.044}_T$	$\mathbf{0.034}_T$	$\mathbf{0.000}_F$	0.118_T	0.064_T	$\mathbf{0.002}_F$	**0.049**	0.580
8-shot	0.026_T	$\mathbf{0.342}_T$	0.497_T	$\mathbf{0.267}_T$	0.186_T	0.035_T	0.229	0.093_T	0.044_T	$\mathbf{0.000}_F$	0.181_T	0.089_T	0.007_F	0.065	0.592

Table 5: *The impact of ICL* on causal relationships tested on *GPT-3.5-Turbo*, where the best |ATE| and task accuracy are marked in bold. The '*T/F*' indicates the statistical significance of the causal relation. The detailed outcomes of the 0/2/4/8-shot can be found in Table 2 and Table 13, 14, 15 in Appendix F, respectively.

Model	|ATE| on 'CoT $\xrightarrow{?}$ Answer' (↑)							|ATE| on 'Instruction $\xrightarrow{?}$ Answer' (↓)							Task
	Add.	Mul.	GSM.	Pro.	FOL.	LQA.	Avg.	Add.	Mul.	GSM.	Pro.	FOL.	LQA.	Avg.	Accuracy
Base	0.006_F	$\mathbf{0.164}_T$	$\mathbf{0.496}_T$	$\mathbf{0.254}_T$	0.034_F	0.040_T	**0.204**	$\mathbf{0.006}_F$	0.040_T	$\mathbf{0.012}_T$	0.411_T	0.089_T	$\mathbf{0.001}_F$	0.111	**0.313**
SFT	$\mathbf{0.008}_F$	0.091_T	0.490_T	0.191_T	$\mathbf{0.054}_F$	0.023_F	0.179	0.021_T	0.072_T	0.098_T	0.381_T	0.086_T	0.200_T	0.156	0.300
DPO	0.006_F	0.026_T	0.417_T	0.119_T	0.049_F	$\mathbf{0.058}_T$	0.138	0.022_T	$\mathbf{0.005}_F$	0.092_T	$\mathbf{0.114}_T$	$\mathbf{0.042}_F$	0.118_T	**0.066**	0.275

Table 6: *The impact of SFT/RLHF* on causal relationships based on *Mistral-7B*, where SFT primarily enhances the causal connection between the instruction and answer, and DPO diminishes this relationship. The detailed outcomes of the Base, SFT, and DPO models can be found in Table 16, 17, and 18 in Appendix F, respectively.

The results reveal that, compared to zero-shot, ICL demonstrations improve causal relationships and enhance task accuracies. Specifically, ICL generally reduces |ATE| in 'Instruction → Answer', but enhances |ATE| in 'CoT → Answer' as indicated by the Avg. |ATE|, resulting in improved causal relationships.

Impact of SFT and RLHF. Supervised fine-tuning (SFT) enables LLMs to follow human instructions (Wei et al., 2022a), while reinforcement learning from human feedback (RLHF) aligns LLMs with human preferences (Christiano et al., 2017; Ouyang et al., 2022). However, recent studies suggest that SFT and RLHF may induce hallucinations (Schulman, 2023; Yang et al., 2023). We hypothesize that they may also affect the causal structures.

We validate our hypothesis by performing causal analysis on three models: a foundation model Mistral-7B-Base [4] (Jiang et al., 2023), an instruct-tuned model Mistral-7B-SFT [5], and an RLHF-tuned model Mistral-7B-DPO [6] (Tunstall et al., 2023). Since the base model cannot follow instructions, we elicit the question-answering behavior using ICL with four demonstrations, and similarly, we use the same demonstrations for the SFT and DPO models.

As Table 6 shows, SFT generally weakens the causal structure, with a smaller Avg. |ATE| on 'CoT → Answer' but a larger Avg. |ATE| on 'Instruction → Answer', indicating that SFT introduces spurious features into the model, thereby causing hallucinations. In contrast, DPO reduces spurious features by weakening the link between the instruction and the answer and reducing the Avg. |ATE| from 0.111 to 0.066. The findings align with the human preference to separate the answers from irrelevant spurious features (Ouyang et al., 2022). However, DPO also weakens the causal connection between CoTs and answers (lower the Avg. |ATE| to 0.138), suggesting a negative side effect of DPO.

5 Conclusion

We conducted causal analyses on LLMs with CoT, revealing the underlying SCM structures, which serve as essential features that can be used to predict the latent behaviors, and further consistency and faithfulness of CoT. Analyses of the relevant factors show that model size has a significant influence on the causal structure, but larger models do not necessarily lead to better SCMs. Popular techniques like ICL, SFT, and RLHF affect causal structures, with ICL strengthening them while SFT and RLHF weakening them.

Our findings underscore the need for further research into effective LLM techniques to strengthen causal structures, with the goal of achieving human-level reasoning ability.

[4] https://huggingface.co/mistralai/Mistral-7B-v0.1

[5] https://huggingface.co/HuggingFaceH4/zephyr-7b-sft-beta

[6] https://huggingface.co/HuggingFaceH4/zephyr-7b-beta

Acknowledgments

We would like to thank the anonymous reviewers for their valuable feedback. This work is funded by the National Natural Science Foundation of China Key Program (Grant No. 62336006) and the Pioneer and "Leading Goose" R&D Program of Zhejiang (Grant No. 2022SDXHDX0003).

Limitations

This study focuses primarily on the analysis of existing model and LLM techniques on their impact on the underlying causal structures, leaving the exploration of new techniques to improve the causal structure in the future. We also focus on the currently popular Generative Pre-Training (GPT) (Radford et al., 2018) language models, setting aside other models such as the Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al., 2018) and the General Language Model (GLM) (Du et al., 2022) for future exploration. This is due to the potentially more intricate causal structures in these models, stemming from their blank-infilling training objective. Furthermore, our research predominantly deals with standard mathematical and logical reasoning, not including areas like common sense and symbolic reasoning within its scope.

Ethical and Broader Impact

The study offers a framework for understanding the decision-making process and reasoning of LLM, which could contribute to greater transparency and accountability in AI systems. It underscores the fact that LLMs can be swayed by unrelated contexts, resulting in biased outcomes. The study implies that the typical techniques employed in LLM might not necessarily improve its reasoning abilities. This could impact the way we train and educate AI models in the future.

References

Joshua Angrist and Guido Imbens. 1995. Identification and estimation of local average treatment effects.

Oliver Bentham, Nathan Stringham, and Ana Marasović. 2024. Chain-of-thought unfaithfulness as disguised accuracy. *arXiv preprint arXiv:2402.14897*.

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

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

Zheng Chu, Jingchang Chen, Qianglong Chen, Weijiang Yu, Tao He, Haotian Wang, Weihua Peng, Ming Liu, Bing Qin, and Ting Liu. 2023. A survey of chain of thought reasoning: Advances, frontiers and future. *arXiv preprint arXiv:2309.15402*.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Denise Dellarosa Cummins. 1995. Naive theories and causal deduction. *Memory & cognition*, 23(5):646–658.

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

Zhengxiao Du, Yujie Qian, Xiao Liu, Ming Ding, Jiezhong Qiu, Zhilin Yang, and Jie Tang. 2022. Glm: General language model pretraining with autoregressive blank infilling. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 320–335.

Amir Feder, Katherine A Keith, Emaad Manzoor, Reid Pryzant, Dhanya Sridhar, Zach Wood-Doughty, Jacob Eisenstein, Justin Grimmer, Roi Reichart, Margaret E Roberts, et al. 2022. Causal inference in natural language processing: Estimation, prediction, interpretation and beyond. *Transactions of the Association for Computational Linguistics*, 10:1138–1158.

York Hagmayer, Steven A Sloman, David A Lagnado, and Michael R Waldmann. 2007. Causal reasoning through intervention. *Causal learning: Psychology, philosophy, and computation*, pages 86–100.

Simeng Han, Hailey Schoelkopf, Yilun Zhao, Zhenting Qi, Martin Riddell, Luke Benson, Lucy Sun, Ekaterina Zubova, Yujie Qiao, Matthew Burtell, et al. 2022. Folio: Natural language reasoning with first-order logic. *arXiv preprint arXiv:2209.00840*.

Sree Harsha Tanneru, Dan Ley, Chirag Agarwal, and Himabindu Lakkaraju. 2024. On the hardness of faithful chain-of-thought reasoning in large language models. *arXiv e-prints*, pages arXiv–2406.

Mary Hegarty. 2004. Mechanical reasoning by mental simulation. *Trends in cognitive sciences*, 8(6):280–285.

David R Heise. 1975. *Causal analysis*. John Wiley & Sons.

Shima Imani, Liang Du, and Harsh Shrivastava. 2023. Mathprompter: Mathematical reasoning using large language models. *arXiv preprint arXiv:2303.05398*.

Guido W Imbens and Donald B Rubin. 2015. *Causal inference in statistics, social, and biomedical sciences*. Cambridge University Press.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *Preprint*, arXiv:2310.06825.

Mingyu Jin, Qinkai Yu, Haiyan Zhao, Wenyue Hua, Yanda Meng, Yongfeng Zhang, Mengnan Du, et al. 2024. The impact of reasoning step length on large language models. *arXiv preprint arXiv:2401.04925*.

Zhijing Jin, Yuen Chen, Felix Leeb, Luigi Gresele, Ojasv Kamal, LYU Zhiheng, Kevin Blin, Fernando Gonzalez Adauto, Max Kleiman-Weiner, Mrinmaya Sachan, et al. 2023a. Cladder: Assessing causal reasoning in language models. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Zhijing Jin, Jiarui Liu, Zhiheng Lyu, Spencer Poff, Mrinmaya Sachan, Rada Mihalcea, Mona Diab, and Bernhard Schölkopf. 2023b. Can large language models infer causation from correlation? *arXiv preprint arXiv:2306.05836*.

Emre Kıcıman, Robert Ness, Amit Sharma, and Chenhao Tan. 2023. Causal reasoning and large language models: Opening a new frontier for causality. *arXiv preprint arXiv:2305.00050*.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. *Preprint*, arXiv:2309.06180.

Tamera Lanham, Anna Chen, Ansh Radhakrishnan, Benoit Steiner, Carson Denison, Danny Hernandez, Dustin Li, Esin Durmus, Evan Hubinger, Jackson Kernion, et al. 2023. Measuring faithfulness in chain-of-thought reasoning. *arXiv preprint arXiv:2307.13702*.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, et al. 2022. Solving quantitative reasoning problems with language models. *Advances in Neural Information Processing Systems*, 35:3843–3857.

Hanmeng Liu, Jian Liu, Leyang Cui, Zhiyang Teng, Nan Duan, Ming Zhou, and Yue Zhang. 2023a. Logiqa 2.0—an improved dataset for logical reasoning in natural language understanding. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*.

Hanmeng Liu, Ruoxi Ning, Zhiyang Teng, Jian Liu, Qiji Zhou, and Yue Zhang. 2023b. Evaluating the logical reasoning ability of chatgpt and gpt-4. *arXiv preprint arXiv:2304.03439*.

Qing Lyu, Shreya Havaldar, Adam Stein, Li Zhang, Delip Rao, Eric Wong, Marianna Apidianaki, and Chris Callison-Burch. 2023. Faithful chain-of-thought reasoning. *arXiv preprint arXiv:2301.13379*.

Quinn McNemar. 1947. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12(2):153–157.

Jovana Mitrovic, Brian McWilliams, Jacob C Walker, Lars Holger Buesing, and Charles Blundell. 2020. Representation learning via invariant causal mechanisms. In *International Conference on Learning Representations*.

OpenAI. 2022. ChatGPT. https://chat.openai.com/.

OpenAI. 2023. GPT-4 Technical Report. *arXiv preprint arXiv:2303.08774*.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. *Preprint*, arXiv:2203.02155.

Liangming Pan, Alon Albalak, Xinyi Wang, and William Yang Wang. 2023. Logic-lm: Empowering large language models with symbolic solvers for faithful logical reasoning. *arXiv preprint arXiv:2305.12295*.

Debjit Paul, Robert West, Antoine Bosselut, and Boi Faltings. 2024. Making reasoning matter: Measuring and improving faithfulness of chain-of-thought reasoning. *arXiv preprint arXiv:2402.13950*.

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

Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. 2017. *Elements of causal inference: foundations and learning algorithms*. The MIT Press.

Jacob Pfau, William Merrill, and Samuel R Bowman. 2024. Let's think dot by dot: Hidden computation in transformer language models. *arXiv preprint arXiv:2404.15758*.

Jing Qian, Hong Wang, Zekun Li, Shiyang Li, and Xifeng Yan. 2022. Limitations of language models in arithmetic and symbolic induction. *arXiv preprint arXiv:2208.05051*.

Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. 2018. Improving language understanding by generative pre-training.

Donald B Rubin. 1974. Estimating causal effects of treatments in randomized and nonrandomized studies. *Journal of educational Psychology*, 66(5):688.

Maarten Sap, Ronan Le Bras, Emily Allaway, Chandra Bhagavatula, Nicholas Lourie, Hannah Rashkin, Brendan Roof, Noah A Smith, and Yejin Choi. 2019. Atomic: An atlas of machine commonsense for if-then reasoning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 3027–3035.

John Schulman. 2023. Reinforcement learning from human feedback: Progress and challenges. In *Berkley Electrical Engineering and Computer Sciences. URL: https://eecs. berkeley. edu/research/colloquium/230419 [accessed 2023-11-15]*.

Steven A Sloman and David Lagnado. 2015. Causality in thought. *Annual review of psychology*, 66:223–247.

Zayne Sprague, Xi Ye, Kaj Bostrom, Swarat Chaudhuri, and Greg Durrett. 2023. Musr: Testing the limits of chain-of-thought with multistep soft reasoning. *arXiv preprint arXiv:2310.16049*.

Oyvind Tafjord, Bhavana Dalvi Mishra, and Peter Clark. 2020. Proofwriter: Generating implications, proofs, and abductive statements over natural language. *arXiv preprint arXiv:2012.13048*.

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

Lewis Tunstall, Edward Beeching, Nathan Lambert, Nazneen Rajani, Kashif Rasul, Younes Belkada, Shengyi Huang, Leandro von Werra, Clémentine Fourrier, Nathan Habib, et al. 2023. Zephyr: Direct distillation of lm alignment. *arXiv preprint arXiv:2310.16944*.

Miles Turpin, Julian Michael, Ethan Perez, and Samuel R Bowman. 2023. Language models don't always say what they think: Unfaithful explanations in chain-of-thought prompting. *arXiv preprint arXiv:2305.04388*.

Robert Vacareanu, Anurag Pratik, Evangelia Spiliopoulou, Zheng Qi, Giovanni Paolini, Neha Anna John, Jie Ma, Yassine Benajiba, and Miguel Ballesteros. 2024. General purpose verification for chain of thought prompting. *arXiv preprint arXiv:2405.00204*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Victor Veitch, Alexander D'Amour, Steve Yadlowsky, and Jacob Eisenstein. 2021. Counterfactual invariance to spurious correlations: Why and how to pass stress tests. *arXiv preprint arXiv:2106.00545*.

Lei Wang, Wanyu Xu, Yihuai Lan, Zhiqiang Hu, Yunshi Lan, Roy Ka-Wei Lee, and Ee-Peng Lim. 2023. Plan-and-solve prompting: Improving zero-shot chain-of-thought reasoning by large language models. *arXiv preprint arXiv:2305.04091*.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2022. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*.

Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. 2022a. Finetuned language models are zero-shot learners. *Preprint*, arXiv:2109.01652.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022b. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837.

T Wu, M Tulio Ribeiro, J Heer, and D Weld. 2021. Polyjuice: Generating counterfactuals for explaining, evaluating, and improving models. In *Joint Conference of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (ACL-IJCNLP 2021)*.

Fangzhi Xu, Qika Lin, Jiawei Han, Tianzhe Zhao, Jun Liu, and Erik Cambria. 2023. Are large language models really good logical reasoners? a comprehensive evaluation from deductive, inductive and abductive views. *arXiv preprint arXiv:2306.09841*.

Linyi Yang, Jiazheng Li, Padraig Cunningham, Yue Zhang, Barry Smyth, and Ruihai Dong. 2021. Exploring the efficacy of automatically generated counterfactuals for sentiment analysis. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 306–316, Online. Association for Computational Linguistics.

Yuqing Yang, Ethan Chern, Xipeng Qiu, Graham Neubig, and Pengfei Liu. 2023. Alignment for honesty. *arXiv preprint arXiv:2312.07000*.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L Griffiths, Yuan Cao, and Karthik Narasimhan. 2023a. Tree of thoughts: Deliberate

problem solving with large language models. *arXiv preprint arXiv:2305.10601*.

Yao Yao, Zuchao Li, and Hai Zhao. 2023b. Beyond chain-of-thought, effective graph-of-thought reasoning in large language models. *arXiv preprint arXiv:2305.16582*.

Evelyn Yee, Alice Li, Chenyu Tang, Yeon Ho Jung, Ramamohan Paturi, and Leon Bergen. 2024. Dissociation of faithful and unfaithful reasoning in llms. *arXiv preprint arXiv:2405.15092*.

Fei Yu, Hongbo Zhang, Prayag Tiwari, and Benyou Wang. 2024. Natural language reasoning, a survey. *ACM Comput. Surv.* Just Accepted.

Matej Zečević, Moritz Willig, Devendra Singh Dhami, and Kristian Kersting. 2023. Causal parrots: Large language models may talk causality but are not causal. *Transactions on Machine Learning Research*.

A Terminology

We provide a brief overview of the concepts about causal analysis, which are mainly from Peters et al. (2017).

Variables: Variables are quantities, characteristics, or properties that can be measured or observed. They can change or vary and typically fall into two categories: dependent variables, which are being tested and measured in an experiment, and independent variables, which are manipulated or controlled in an experiment. In this paper, the three variables refer to the three parts involved in the CoT reasoning.

Spurious Correlation: Spurious correlation refers to a mathematical relationship in which two or more variables are not causally related to each other, yet it may be wrongly inferred that they are, due to either coincidence or the presence of a certain third, unseen factor (referred to as a "confounding factor"). The statistical training of LLMs introduce spurious correlations because without treatment experiments, the spurious correlation and causal correlation cannot be inferred from the training data alone.

Causal Relation: A causal relation between two variables exists if the occurrence of the first causes the other. The first variable is called the cause, and the second variable is called the effect. A causal relationship is often established by methodically manipulating the cause and observing the effect. In this paper, we use systematic treatment experiments to detect the significance of the causal relations.

Causal Structure Model (SCM): A causal structure model (also known as a causal model or structural causal model) is a conceptual model that describes the causal mechanisms of a system. The model is usually formalized as directed acyclic graph (DAG), where the nodes represent the variables and the edges represent causal relationships between the variables. This model helps in understanding how a system works and predicting the effects of interventions.

Causal Chain: A sequence of variables, each one triggering the next. In a causal chain, a variable occurs due to some cause, which itself is the effect of some other cause, and so on. It helps to trace the root cause of any variable. In the context of CoT reasoning, the expected causal relations between three variables form a causal chain: the task description (instruction, variable Z) decides

the CoT reasoning steps (variable X), and the reasoning steps decide the final answer (variable Y). Such a causal chain exists in the correct logical and mathematical reasoning process.

B Causal Analysis

Causal analysis is a method that is used to identify and understand the causes and effects of different actions, situations, or decisions. It involves examining the reasons or causes behind a certain occurrence and the outcomes that may arise from it (Heise, 1975; Imbens and Rubin, 2015; Feder et al., 2022). Causal models in different structures may induce the same observational distribution but different intervention distributions (Peters et al., 2017). Interventions thus can be used to differentiate among the potential causal structures that are compatible with an observation (Hagmayer et al., 2007; Pearl, 2009). We study CoT by its causal relationship to the model decisions, using interventions to test the significance of the cause-effect relations between CoTs/instructions and answers.

The basic concepts of SCM and confounder are described as follows.

Definition B.1 (Structural Causal Model) *A simple SCM $\mathcal{G}$ with a graph $X \to Y$ consists of two random variables X and Y, following assignments*

$$\begin{aligned} X &:= N_X, \\ Y &:= f_Y(X, N_Y), \end{aligned} \tag{5}$$

where the noise N_Y is independent of N_X.

*We refer to the random variables X as the **cause** and Y as the **effect**. The graph $X \to Y$ indicates that X is a direct cause of Y.*

Definition B.2 (Confounder of Variables) *An SCM $\mathcal{G}$ with a graph consists of three random variables X, Y, and Z, following assignments*

$$\begin{aligned} X &:= f_X(Z, N_{XZ}), \\ Y &:= f_Y(Z, N_{YZ}), \end{aligned} \tag{6}$$

where the noise N_{YZ} is independent of N_{XZ}.

*We refer to the random variable Z as the **confounder** of the random variables X and Y.*

B.1 Design of CoT Interventions

The golden CoT is designed to enhance accuracy by offering clear hints toward the correct solution, whereas the random CoT is likely to reduce accuracy due to the inclusion of distorted information (such as incorrect numbers for math tasks or misleading statements for logic problems), which can misguide the reasoning process toward an incorrect conclusion.

The golden CoT does not leave much room for considering different options. On the other hand, random CoT presents various alternatives. We experiment with several techniques to disrupt the CoT steps. For example, we negate one third of the CoT statements at different points – beginning, middle, and end – and observe that altering the beginning or middle parts did not significantly affect the outcomes. Consequently, we focus on the end part.

We also test other disturbance strategies for the CoT, like mixing up the names of entities or altering the sequence of reasoning steps. These methods typically cause the model to initiate a new line of reasoning, thus disregarding the manipulated CoT. In contrast, our chosen method of intervention successfully disrupts the CoT while still keeping the model's dependence on the given reasoning pathway intact.

B.2 Design of Random Instruction

Role	Instruction
Original	Please act as a math teacher and solve the math problem step by step.
	Intervened Instructions:
Chef	Imagine you are a chef in a bustling kitchen, and you need to tackle this math problem as if it were a recipe. Break down the solution into clear, step-by-step instructions.
Detective	Imagine you are a detective unraveling a mystery. Solve the problem meticulously, step by step, as you would piece together clues in an investigation.
Judge	I need you to take on the role of a judge and adjudicate the math problem, providing a detailed step-by-step resolution.
Artist	Imagine you are an artist, and approach solving the math problem with creativity and flair, breaking it down into steps.

Table 7: GSM8K Original and Intervened Instructions

To intervene in the instructions within the prompt, we adhere to the following principles: 1) There is a controllable variable to adjust the direction of the intervention. 2) There is a clear distinction between the pre- and post-intervention expressions. 3) The intervened instructions should still guide the model in derive the answer through CoT. Based on these principles, we instruct the GPT-4 to paraphrase the original instructions and control the

target distribution of the paraphrase through predefined arbitrary roles (professions). As an example, we present the original and intervened instructions for GSM8K in Table 7.

C Experimental Settings

C.1 Prompts and Templates

We use general prompts and templates for experiments with *Direct* answering and zero-shot *CoT*. To facilitate answer matching, we instructed the model to respond using a template format, based on the approach of prompt modifications by Pan et al. (2023). The prompts for each task are as follows.

Addition (Direct):

Please act as a math teacher and solve the math problem. Please directly answer with the format "The answer is «answer»" without any other information.

What is the sum of {{number1}} and {{number2}}?

Addition (CoT):

Please act as a math teacher and solve the addition problem in the given template.
####
Question:
What is the sum of «number1» and «number2»?
Reasoning:
Let's add the two numbers digit by digit.
1. The ones place: «digit1»
2. The tens place: «digit2»
«other digits»
Answer:
Therefore, the final computed sum is «answer».
####
Question:
What is the sum of {{number1}} and {{number2}}?
Reasoning:

GSM8K (Direct):

Please act as a math teacher and solve the math problem. Please directly answer with the format "The answer is «answer»" without any other information.

{{question}}

GSM8K (CoT):

Please act as a math teacher and solve the math problem step by step.
Question:
{{question}}
Reasoning:
Let's think step by step.

Multiplication (Direct):

Please act as a math teacher and solve the math problem. Please directly answer with the format "The answer is «answer»" without any other information.

What is the product of {{number1}} and {{number2}}?

Multiplication (CoT):

Please act as a math teacher and solve the product problem in the given template.
####
Question:
What is the product of «number1 such as 27» and «number2 such as 153»?
Reasoning:
Let's think step by step. «number2 153 has three digits, so that we can reason in three steps.»
1. «Multiply number1 27 by the ones place digit 3 of number2 153»
2. «Multiply number1 27 by the tens place digit 50 of number2 153»
«other digits of number2 if it has»
Answer:
Now, sum all the step results: «sum of the results».
So, the final computed product is «answer».
####
Question:
What is the product of {{number1}} and {{number2}}?
Reasoning:

ProofWriter/FOLIO/LogiQA (Direct):

Your goal is to solve the logical reasoning problem. Given a context and a question, directly answer with the format "The correct option is: A/B/C" without any other information.
####
Context:
{{context}}

Question:
{{question}}
Options:
{{options}}

Instruction: ## Answer:

ProofWriter/FOLIO/LogiQA (CoT):

Please act as a math teacher and reason step by step to solve the logical reasoning problem. Given a context and a question, explain your reasoning process and give the answer with the format "The correct option is: A/B/C".
####
Context:
{{context}}

Question:
{{question}}
Options:
{{options}}

Instruction:
Reasoning:

For LogiQA, the correct option is: A/B/C/D.

Readers may wonder why the CoT prompts for Addition and Multiplication look so different from others and why we do not just simply use "let's think step by step". In practice, the instruction "let's think step by step" does not consistently trigger a step-by-step reasoning process for Addition and Multiplication tasks on both GPT-3.5-turbo and GPT-4. Therefore, we tested 20 samples each on GPT-3.5-turbo and GPT-4 to derive the most common templates for regulating the responses.

C.2 Normalize Reasoning Steps

We use GPT-3.5-turbo to normalize the generated reasoning steps for Addition and Multiplication, as the following prompts show.

Addition

Please convert the natural language described reasoning steps into formal expressions as the examples. Please put the carry 1 at the last of the addition of each step.
####
Reasoning Steps:
Let's add the two numbers digit by digit.
1. The ones place: 0 + 0 = 0
2. The tens place: 2 + 9 = 11 (carry over the 1)
3. The hundreds place: 7 + 8 + 1 = 16 (carry over the 1)
4. The millions place: 1 + 2 + 1 = 4
Formal Expressions:
1. 0 + 0 = 0
2. 2 + 9 = 11
3. 7 + 8 + 1 (carry) = 16
4. 1 + 2 + 1 (carry) = 4
####
... (other examples)
####
Reasoning Steps:
reason
Formal Expressions:

Multiplication

Please convert the natural language described reasoning steps into formal expressions as the examples.
####
Reasoning Steps:
Let's think step by step. 305 has three digits, so that we can reason in three steps.
1. Multiply 487 by the ones place digit 5 of 305. The result is 2435.
2. Multiply 487 by the tens place digit (0*10) of 305. The result is 0.
3. Multiply 487 by the hundreds place digit (3*100) of 305. The result is 146100.
Formal Expressions:
1. 487 * 5 = 2435
2. 487 * 0 = 0
3. 487 * 300 = 146100
####
... (other examples)
####
Reasoning Steps:
reason
Formal Expressions:

The conversion from verbal reasoning steps to formal expressions is fairly straightforward. We manually reviewed 50 random samples for each task and observed accurate conversions.

In the intervention experiment, we randomly sample an instruction corresponding to a role for replacement and then observe the changes in the output of the LLM. The prompt provided to GPT-4 to generate the instructions is as follows:

Intervene on Instructions

Below is a prompt to instruct LLM to tackle a problem, initially framed as a math teacher solving the issue. Your task is to rephrase the prompt with these adjustments:
1. Request the LLM to assume the role of {{role}} while solving the problem.
2. Alter the sentence structure to enhance differentiation.
3. Add more interference to the prompt and make the prompt more different.
4. Retain any specified output format requirements unchanged.
5. Output the paraphrased prompt directly, do not incorporate other information.
Prompt
{{prompt}}

Paraphrased prompt:

D CoT and Its Effectiveness

We conduct an empirical examination of CoT in six tasks in terms of task accuracy and reasoning errors. The results show that CoT does not consistently improve task performance, with instances where incorrect CoTs lead to correct answers and vice versa, indicating a potential spurious correlation between the CoTs and the answers.

D.1 Variable Effects of CoT

We assess the effectiveness of CoT by contrasting it with direct answering that does not involve step-by-step reasoning, as shown in Table 8. The main observations are as follows.

CoT impairs performance in basic arithmetic tasks. LLMs have been found to successfully pass college entrance level exams (OpenAI, 2023), which are considerably more challenging than primary school arithmetic from a human point of view. However, as the figure illustrates, CoT in basic arithmetic tasks reveals relatively low accuracies (below 0.55) on GPT-3.5-turbo. Conversely, direct answers in Addition achieve significantly higher accuracies (above 0.95).

CoT enhances performance in complex reasoning tasks. In the GSM8K math word problem,

LLM	Method	Addition (6 digits)	Addition (9 digits)	Multiplication (2 digits)	Multiplication (3 digits)	GSM8K	ProofWriter	FOLIO	LogiQA	Avg.
Llama2-7B-Chat	Direct	**0.618** *	**0.108** *	**0.242** *	**0.008**	0.060	0.382	0.392	**0.390**	**0.275**
	CoT	0.136	0.016	0.080	0.002	**0.270** *	**0.398**	**0.480**	0.372	0.219
Llama2-70B-Chat	Direct	**0.826** *	**0.552** *	**0.488** *	**0.044**	0.166	**0.527**	0.495	**0.562**	**0.458**
	CoT	0.592	0.452	0.146	0.032	**0.562** *	0.523	**0.500**	0.548	0.419
GPT-3.5-Turbo	Direct	**0.962** *	**0.958** *	**1.000** *	**0.542** *	0.300	0.277	0.505	**0.548** *	**0.637**
	CoT	0.674	0.372	0.848	0.450	**0.748** *	**0.518** *	**0.574**	0.465	0.581
GPT-4	Direct	0.996	0.986	0.964	0.488	0.496	0.552	**0.701**	**0.702**	0.736
	CoT	**0.998**	**0.990**	**0.990**	**0.816** *	**0.946** *	**0.708** *	0.686	0.688	**0.853**

Table 8: LLMs with CoT show mixed results, improving accuracy in some tasks while reducing it in others. The asterisk '*' indicates a statistical significance of p-value < 0.01 in McNemar's test. All experiments are in zero-shot settings.

LLM	CoT→Answer	Error	Simple Task		Complex Task			
			Add.(6 digits)	**Mult.**(3 digits)	**GSM8K**	**ProofWriter**	**FOLIO**	**LogiQA**
GPT-3.5-Turbo	✓→✓	-	0.026	0.446	0.760	0.260	0.455	0.330
	✓→✗	*type 1*	*0.000*	*0.440*	*0.000*	*0.000*	*0.000*	*0.010*
	✗→✓	*type 2*	*0.648*	*0.004*	*0.000*	*0.280*	*0.125*	*0.115*
	✗→✗	type 3	0.326	0.110	0.240	0.460	0.420	0.545
	consistency error	type 1,2	0.648	0.444	0.000	0.280	0.125	0.125
GPT-4	✓→✓	-	0.254	0.804	0.945	0.640	0.630	0.620
	✓→✗	*type 1*	*0.000*	*0.166*	*0.000*	*0.000*	*0.000*	*0.000*
	✗→✓	*type 2*	*0.744*	*0.012*	*0.000*	*0.060*	*0.070*	*0.010*
	✗→✗	type 3	0.002	0.018	0.055	0.300	0.300	0.370
	consistency error	type 1,2	0.744	0.178	0.000	0.060	0.070	0.010

Table 9: *Consistency errors* produced by LLMs with CoT, where the positive error rates are highlighted in *red*.

CoT consistently improves performance compared to direct answering, highlighting the effectiveness of step-by-step reasoning in solving more complex math problems. In logical reasoning problems, the improvement by CoT is relatively minor, but still consistent across FOLIO and ProofWriter. However, CoT struggles with real-world logical problems such as LogiQA, indicating a discrepancy.

D.2 Inconsistent Behaviors of CoT

CoT performs poorly in basic arithmetic calculations but excels in complex mathematical and logical reasoning tasks, contradicting our intuition. We go into this by evaluating the confusion matrices of the CoT steps and answers as in Table 9, finding even more puzzling behaviors of CoT.

Incorrect CoTs result in correct answers. A considerable portion (over 60%) of Addition samples exhibit this unusual behavior, where the reasoning steps are incorrect but yield the correct answers. This pattern persists even with larger LLMs, where the proportion increases to 74% in GPT-4, suggesting that the problem may not be solved simply by enlarging the model.

A notable proportion of logical reasoning samples also display this peculiar correlation. A manual review in ProofWriter reveals that 28% of the responses from GPT-3.5-turbo have correct answers but incorrect CoTs, which include fabricated facts and faulty inferences. The same pattern is also observed in FOLIO and LogiQA, albeit at a lower frequency, implying that the phenomenon is common in different tasks.

Correct CoTs result in incorrect answers. A significant proportion of Multiplication samples generate incorrect answers with correct reasoning steps. In these instances, we find that the answer steps either fail to gather the intermediate results from the reasoning steps or fail to correctly sum the results. This type of error is seldom seen in other tasks like GSM8K and logical problems because the answer steps in these tasks are straightforward and thus the models are less likely to fail.

D.3 Error Analysis

LLMs with CoT fail on Addition and Multiplication mainly for two reasons: digit collection error and calculation error. The collection error occurs when a reasoning step locates a wrong digit for the calculation, such as step 3 in the failure case of Addition in Figure 3. The hundreds place for 625126 and 542611 is 1 and 6 but the step locates to 5 and 6 thereby causing the error. The calculation error could happen in any step involve a digital calculation, like the answer in the failure case of Multiplication in the figure, where the summation of 3432 + 17160 + 514800 gives 533392 but it should be 535392.

In ProofWriter, the model may identify the correct rule for reasoning through superficial keyword matching rather than logical reasoning, as shown in Figure 3. Even worse, the CoT not only makes

Addition (from GPT-3.5-Turbo)

Please act as a math teacher and solve the addition problem in the given template. ...
####
Question:
What is the sum of 625126 and 542611?
Reasoning:
Let's add the two numbers digit by digit.
1. The ones place: 6 + 1 = 7
2. The tens place: 2 + 1 = 3
3. The hundreds place: 5 + 6 = 11. Write down the 1 and carry over the 10.
4. The thousands place: 2 + 4 + 1 (carried over) = 7
5. The ten thousands place: 6 + 5 = 11. Write down the 1 and carry over the 10.
6. The hundred thousands place: 5 + 2 + 1 (carried over) = 8
7. The millions place: 6
Answer:
Therefore, the final computed sum is 1163737.
(Both the process and final result are incorrect, the correct answer should be 1167737.)

Multiplication (from GPT-4)

Please act as a math teacher and solve the product problem in the given template. ...
####
Question:
What is the product of 858 and 624?
Reasoning:
Let's think step by step. 624 has three digits, so that we can reason in three steps.
1. Multiply 858 by the ones place digit 4 of 624, which is 3432.
2. Multiply 858 by the tens place digit 2 of 624, which is 1716 after the result is multiplied by 10 (since 2 is in the tens place) which results in 17160.
3. Multiply 858 by the hundreds place digit 6 of 624, which is 5148 after the result is multiplied by 100 (since 6 is in the hundreds place) which results in 514800.
Answer:
Now, sum all the step results: 3432 + 17160 + 514800 = 533392.
So, the final computed product is 533392.
(Error in the total calculation, the correct sum should be 535392.)

ProofWriter (from GPT-4)

Please act as a math teacher and reason step by step to solve the logical reasoning problem.
...
####
Context:
Anne is furry. Anne is kind. Anne is nice. Anne is young. Bob is big. Bob is furry. Charlie is nice. Charlie is young. Dave is cold. Dave is kind. Dave is young. All cold, big people are furry. All big people are rough. All kind, big people are rough. If someone is nice and rough then they are kind. If Charlie is kind then Charlie is big. If someone is rough and kind then they are cold. Nice people are big. All rough, furry people are cold. Cold, nice people are big.
Question:
Based on the above information, is the following statement true, false, or unknown? Charlie is furry.
Reasoning:
Let's break down the information given:
1. Charlie is nice and young.
2. If Charlie is kind then Charlie is big.
3. Nice people are big. So, Charlie is big.
4. If someone is nice and rough then they are kind. So, Charlie is kind.
5. If someone is rough and kind then they are cold. So, Charlie is cold.
6. All cold, big people are furry. So, Charlie is furry.
Therefore, based on the given information, Charlie is furry.
The correct option is: A) True.
(Step 2 is mentioned but not actually used, step 4 and 5 miss their precondition of 'Charlie is rough', which can be derived from 'All big people are rough'. The right answer is A.)

Figure 3: Three examples of CoT mistakes, where either the CoT is incorrect but the answer is correct, or the other way around. The red highlights the incorrect steps, with explanations provided at the end of each.

incorrect causal reasoning, but invents factual information that is not provided in the context, yet the invented reasoning still leads to the correct answers.

E Discussion

What is the difference between Llama2 and Mistral series? The 7B models including Llama2-7B and Mistral-7B perform poorly on Addition and Multiplication tasks, where the accuracies are less than 0.2. Among these tasks, Llama2 performs better on Addition, while Mistral performs better on Multiplication. Generally, the implied SCM types for Mistral-7B-DPO surpass those for Llama2-7B-Chat.

Do ChatGPT and GPT-4 perform perfectly on GSM8K like human reasoners? While both ChatGPT and GPT-4 suggest the ideal type I SCM on GSM8K, Tables 2 and 10 indicate that the Average Treatment Effect (ATE) is not perfectly zero. This implies that although ChatGPT and GPT-4 perform similarly to a human in reasoning, they are not perfect due to their statistical nature.

Future Directions. In this study, we focus on the basic CoT, leaving the analysis of other alternatives like tree-of-thought and graph-of-thought to the future. Furthermore, our analysis is on a coarse-grained reasoning process, where it is simplified into only three random variables. More detailed analysis of fine-grained structures can be a direction for future work. The causal structures in LLMs could potentially be enhanced during the training of LLMs, for example, using counterfactual examples (Mitrovic et al., 2020; Wu et al., 2021; Yang et al., 2021) or causal regulation (Veitch et al., 2021), which can be worth exploration in future work.

F Detailed Results

More detailed results are presented in following tables.

Intervention	GPT-4					
	Addition	**Multiplication**	**GSM8K**	**ProofWriter**	**FOLIO**	**LogicQA**
CoT	0.998	0.816	0.946	0.708	0.686	0.688
Test: If CoT causes Answer given constant Instruction?						
Controlled (w/ default setting)	1.000	0.858	0.948	0.708	0.686	0.685
Treated (w/ golden CoT)	*+0.000*	*+0.062 **	*+0.052 **	*+0.285 **	-	-
Treated (w/ random CoT)	*-0.006*	*-0.818 **	*-0.016*	*-0.082 **	*-0.049*	*-0.002*
CoT $\xrightarrow{?}$ Answer	F	T	T	T	F	F
Test: If Instruction causes Answer given constant CoT?						
Controlled (w/ default setting)	1.000	0.858	0.948	0.708	0.686	0.685
Treated (w/ random instruction)	*-0.002*	*-0.044*	*-0.002*	*+0.000*	*+0.000*	*+0.003*
Treated (w/ random bias)	*-0.174 **	*-0.010*	*-0.002*	*-0.005*	*-0.015*	*-0.010*
Controlled (w/ golden CoT)	1.000	0.920	1.000	0.993	-	-
Treated (w/ random instruction)	*-0.002*	*-0.008*	*+0.000*	*-0.007*	-	-
Treated (w/ random bias)	*-0.154 **	*-0.026*	*-0.006*	*-0.060 **	-	-
Instruction $\xrightarrow{?}$ Answer	T	F	F	T	F	F
Implied SCM Type	II	I	I	III	IV	IV

Table 10: *Identification of causal structures* in tasks running on *GPT-4*. The symbol '*' denotes the average treatment effect (ATE) which is significant with a p-value less than 0.01.

Intervention	Llama2-7B-Chat					
	Addition	**Multiplication**	**GSM8K**	**ProofWriter**	**FOLIO**	**LogicQA**
CoT	0.136	0.002	0.270	0.398	0.480	0.372
Test: If CoT causes Answer given constant Instruction?						
Controlled (w/ default setting)	0.120	0.002	0.280	0.400	0.510	0.335
Treated (w/ golden CoT)	*+0.052*	*+0.010*	*+0.518 **	*+0.138 **	-	-
Treated (w/ random CoT)	*-0.014*	*-0.002*	*-0.164 **	*-0.020*	*-0.020*	*+0.005*
CoT $\xrightarrow{?}$ Answer	F	F	T	T	F	F
Test: If Instruction causes Answer given constant CoT?						
Controlled (w/ default setting)	0.120	0.002	0.280	0.400	0.510	0.335
Treated (w/ random instruction)	*-0.052 **	*+0.000*	*-0.022*	*-0.023*	*-0.064*	*-0.012*
Treated (w/ random bias)	*-0.116 **	*+0.000*	*-0.084 **	*-0.283 **	*-0.324 **	*-0.100 **
Controlled (w/ golden CoT)	0.172	0.012	0.798	0.538	-	-
Treated (w/ random instruction)	*-0.076 **	*+0.016*	*+0.008*	*-0.060*	-	-
Treated (w/ random bias)	*-0.172 **	*+0.000*	*-0.070 **	*-0.480 **	-	-
Instruction $\xrightarrow{?}$ Answer	T	F	T	T	T	T
Implied SCM Type	**II**	**IV**	**III**	**III**	**II**	**II**

Table 11: *Identification of causal structures* in tasks running on *Llama2-7B-Chat*. The symbol '*' denotes the average treatment effect (ATE) which is significant with a p-value less than 0.01.

Intervention	Llama2-70B-Chat					
	Addition	**Multiplication**	**GSM8K**	**ProofWriter**	**FOLIO**	**LogicQA**
CoT	0.592	0.032	0.562	0.523	0.500	0.548
Test: If CoT causes Answer given constant Instruction?						
Controlled (w/ default setting)	0.458	0.016	0.552	0.525	0.510	0.543
Treated (w/ golden CoT)	*-0.126 **	*+0.180 **	*+0.356 **	*+0.092 **	-	-
Treated (w/ random CoT)	*-0.062 **	*+0.012*	*-0.032*	*-0.025 **	*-0.044*	*+0.002*
CoT $\xrightarrow{?}$ Answer	T	T	T	T	F	F
Test: If Instruction causes Answer given constant CoT?						
Controlled (w/ default setting)	0.458	0.016	0.552	0.525	0.510	0.543
Treated (w/ random instruction)	*-0.018*	*+0.000*	*+0.002*	*-0.005*	*+0.005*	*-0.003*
Treated (w/ random bias)	*-0.036*	*+0.018*	*-0.024*	*-0.008*	*+0.000*	*+0.000*
Controlled (w/ golden CoT)	0.332	0.196	0.908	0.617	-	-
Treated (w/ random instruction)	*+0.054*	*-0.096 **	*-0.048 **	*-0.037*	-	-
Treated (w/ random bias)	*-0.304 **	*-0.052 **	*-0.224 **	*-0.208 **	-	-
Instruction $\xrightarrow{?}$ Answer	T	T	T	T	F	F
Implied SCM Type	**III**	**III**	**III**	**III**	**IV**	**IV**

Table 12: *Identification of causal structures* in tasks running on *Llama2-70B-Chat*. The symbol '*' denotes the average treatment effect (ATE) which is significant with a p-value less than 0.01.

Intervention	GPT-3.5-Turbo (2-Shot)						
	Addition	**Multiplication**	**GSM8K**	**ProofWriter**	**FOLIO**	**LogicQA**	**Avg. $\lvert ATE \rvert$**
CoT	0.576	0.650	0.754	0.553	0.549	0.508	-
Test: If CoT causes Answer given constant Instruction?							
Controlled (w/ default setting)	0.638	0.634	0.754	0.537	0.569	0.515	-
Treated (w/ golden CoT)	*-0.128 **	*+0.042 **	*+0.246 **	*+0.263 **	-	-	0.170
Treated (w/ random CoT)	*+0.062 **	*-0.634 **	*-0.748 **	*-0.245 **	*-0.265 **	*-0.043 **	0.333
CoT $\xrightarrow{?}$ Answer	T	T	T	T	T	T	0.251
Test: If Instruction causes Answer given constant CoT?							
Controlled (w/ default setting)	0.638	0.634	0.754	0.537	0.569	0.515	-
Treated (w/ random instruction)	*-0.214 **	*-0.004*	*+0.000*	*+0.000*	*-0.029*	*-0.002*	0.042
Treated (w/ random bias)	*-0.112 **	*+0.066 **	*+0.000*	*-0.107 **	*-0.088 **	*-0.010*	0.064
Controlled (w/ golden CoT)	0.510	0.676	1.000	0.800	-	-	-
Treated (w/ random instruction)	*-0.078 **	*+0.000*	*+0.000*	*+0.000*	-	-	0.020
Treated (w/ random bias)	*-0.012*	*+0.070 **	*+0.000*	*-0.362 **	-	-	0.111
Instruction $\xrightarrow{?}$ Answer	T	T	F	T	T	F	0.059
Implied SCM Type	**III**	**III**	**I**	**III**	**III**	**I**	-

Table 13: *Identification of causal structures* in tasks running on *GPT-3.5-Turbo with 2-Shot*.

Intervention	GPT-3.5-Turbo (4-Shot)						
	Addition	**Multiplication**	**GSM8K**	**ProofWriter**	**FOLIO**	**LogicQA**	**Avg. |ATE|**
CoT	0.380	0.670	0.744	0.568	0.618	0.500	-
Test: If CoT causes Answer given constant Instruction?							
Controlled (w/ default setting)	0.354	0.672	0.744	0.568	0.603	0.502	-
Treated (w/ golden CoT)	*+0.016*	*+0.000*	*+0.256 **	*+0.270 **	-	-	0.136
Treated (w/ random CoT)	*+0.012*	*-0.672 **	*-0.740 **	*-0.232 **	*-0.235 **	*-0.022*	0.319
CoT $\xrightarrow{?}$ Answer	F	T	T	T	T	F	0.227
Test: If Instruction causes Answer given constant CoT?							
Controlled (w/ default setting)	0.354	0.672	0.744	0.568	0.603	0.502	-
Treated (w/ random instruction)	*+0.074 **	*+0.010*	*+0.000*	*-0.007*	*+0.005*	*-0.002*	0.016
Treated (w/ random bias)	*+0.008*	*+0.058 **	*+0.000*	*-0.075 **	*-0.123 **	*-0.002*	0.044
Controlled (w/ golden CoT)	0.370	0.672	1.000	0.838	-	-	-
Treated (w/ random instruction)	*+0.040*	*+0.002*	*+0.000*	*-0.040*	-	-	0.021
Treated (w/ random bias)	*+0.052*	*+0.066 **	*+0.000*	*-0.348 **	-	-	0.117
Instruction $\xrightarrow{?}$ Answer	T	T	F	T	T	F	0.049
Implied SCM Type	**II**	**III**	**I**	**III**	**III**	**IV**	-

Table 14: *Identification of causal structures* in tasks running on *GPT-3.5-Turbo with 4-Shot*.

Intervention	GPT-3.5-Turbo (8-Shot)						
	Addition	**Multiplication**	**GSM8K**	**ProofWriter**	**FOLIO**	**LogicQA**	**Avg. |ATE|**
CoT	0.456	0.636	0.772	0.567	0.608	0.512	-
Test: If CoT causes Answer given constant Instruction?							
Controlled (w/ default setting)	0.414	0.650	0.772	0.577	0.603	0.512	-
Treated (w/ golden CoT)	*+0.006*	*+0.036 **	*+0.228 **	*+0.262 **	-	-	0.133
Treated (w/ random CoT)	*+0.046 **	*-0.648 **	*-0.766 **	*-0.272 **	*-0.186 **	*-0.035 **	0.326
CoT $\xrightarrow{?}$ Answer	T	T	T	T	T	T	0.229
Test: If Instruction causes Answer given constant CoT?							
Controlled (w/ default setting)	0.414	0.650	0.772	0.577	0.603	0.512	-
Treated (w/ random instruction)	*-0.016*	*-0.016*	*+0.000*	*-0.040 **	*-0.025*	*+0.005*	0.017
Treated (w/ random bias)	*+0.058*	*+0.076 **	*+0.000*	*-0.190 **	*-0.152 **	*-0.008*	0.081
Controlled (w/ golden CoT)	0.420	0.686	1.000	0.838	-	-	-
Treated (w/ random instruction)	*+0.000*	*+0.004*	*+0.000*	*-0.102 **	-	-	0.027
Treated (w/ random bias)	*+0.076 **	*+0.080 **	*+0.000*	*-0.393 **	-	-	0.137
Instruction $\xrightarrow{?}$ Answer	T	T	F	T	T	F	0.065
Implied SCM Type	**III**	**III**	**I**	**III**	**III**	**I**	-

Table 15: *Identification of causal structures* in tasks running on *GPT-3.5-Turbo with 8-Shot*.

Intervention	Mistral-7B-Base (4-Shot)						
	Addition	**Multiplication**	**GSM8K**	**ProofWriter**	**FOLIO**	**LogicQA**	**Avg. |ATE|**
CoT	0.002	0.176	0.434	0.403	0.412	0.450	-
Test: If CoT causes Answer given constant Instruction?							
Controlled (w/ default setting)	0.002	0.176	0.434	0.510	0.422	0.443	-
Treated (w/ golden CoT)	*+0.000*	*+0.152 **	*+0.566 **	*+0.345 **	-	-	0.266
Treated (w/ random CoT)	*+0.012*	*-0.176 **	*-0.426 **	*-0.163 **	*+0.034*	*-0.040 **	0.142
CoT $\xrightarrow{?}$ Answer	F	T	T	T	F	T	0.204
Test: If Instruction causes Answer given constant CoT?							
Controlled (w/ default setting)	0.002	0.176	0.434	0.510	0.422	0.443	-
Treated (w/ random instruction)	*-0.002*	*-0.042 **	*+0.000*	*-0.135 **	*-0.074*	*-0.002*	0.043
Treated (w/ random bias)	*-0.002*	*-0.012*	*-0.010*	*-0.448 **	*-0.103 **	*+0.000*	0.096
Controlled (w/ golden CoT)	0.002	0.328	1.000	0.855	-	-	-
Treated (w/ random instruction)	*-0.002*	*-0.090 **	*-0.026 **	*-0.343 **	-	-	0.115
Treated (w/ random bias)	*+0.016*	*-0.014*	*-0.012*	*-0.717 **	-	-	0.190
Instruction $\xrightarrow{?}$ Answer	F	T	T	T	T	F	0.111
Implied SCM Type	**IV**	**III**	**III**	**III**	**II**	**I**	-

Table 16: *Identification of causal structures* in tasks running on *Mistral-7B-Base*.

Intervention	Mistral-7B-SFT (4-Shot)						
	Addition	**Multiplication**	**GSM8K**	**ProofWriter**	**FOLIO**	**LogicQA**	**Avg. |ATE|**
CoT	0.008	0.046	0.492	0.375	0.490	0.388	-
Test: If CoT causes Answer given constant Instruction?							
Controlled (w/ default setting)	0.030	0.052	0.484	0.377	0.240	0.460	-
Treated (w/ golden CoT)	*-0.002*	*+0.134 **	*+0.510 **	*+0.378 **	-	-	0.256
Treated (w/ random CoT)	*+0.014*	*-0.048 **	*-0.470 **	*+0.003*	*+0.054*	*-0.023*	0.102
CoT $\xrightarrow{?}$ Answer	F	T	T	T	F	F	0.179
Test: If Instruction causes Answer given constant CoT?							
Controlled (w/ default setting)	0.030	0.052	0.484	0.377	0.240	0.460	-
Treated (w/ random instruction)	*-0.012*	*-0.038 **	*-0.088 **	*-0.195 **	*-0.088*	*-0.295 **	0.119
Treated (w/ random bias)	*-0.030 **	*-0.030 **	*-0.046 **	*-0.088 **	*-0.083 **	*-0.105 **	0.064
Controlled (w/ golden CoT)	0.028	0.186	0.994	0.755	-	-	-
Treated (w/ random instruction)	*-0.016*	*-0.124 **	*-0.168 **	*-0.575 **	-	-	0.221
Treated (w/ random bias)	*-0.026 **	*-0.096 **	*-0.090 **	*-0.665 **	-	-	0.219
Instruction $\xrightarrow{?}$ Answer	T	T	T	T	T	T	0.156
Implied SCM Type	**II**	**III**	**III**	**III**	**II**	**II**	-

Table 17: *Identification of causal structures* in tasks running on *Mistral-7B-SFT*.

Intervention	Mistral-7B-DPO (4-Shot)						
	Addition	**Multiplication**	**GSM8K**	**ProofWriter**	**FOLIO**	**LogicQA**	**Avg. |ATE|**
CoT	0.000	0.012	0.326	0.322	0.520	0.470	-
Test: If CoT causes Answer given constant Instruction?							
Controlled (w/ default setting)	0.000	0.010	0.324	0.310	0.397	0.485	-
Treated (w/ golden CoT)	*+0.004*	*+0.048 **	*+0.522 **	*+0.238 **	-	-	0.203
Treated (w/ random CoT)	*+0.008*	*-0.006*	*-0.312*	*+0.000*	*-0.049*	*-0.058 **	0.072
CoT $\xrightarrow{?}$ Answer	F	T	T	T	F	T	0.138
Test: If Instruction causes Answer given constant CoT?							
Controlled (w/ default setting)	0.000	0.010	0.324	0.310	0.397	0.485	-
Treated (w/ random instruction)	*+0.034 **	*+0.002*	*-0.066 **	*-0.067 **	*-0.049*	*-0.090 **	0.051
Treated (w/ random bias)	*+0.002*	*-0.006*	*-0.028*	*-0.035*	*-0.034*	*-0.145 **	0.042
Controlled (w/ golden CoT)	0.004	0.058	0.846	0.548	-	-	-
Treated (w/ random instruction)	*+0.034 **	*+0.002*	*-0.162 **	*-0.153 **	-	-	0.088
Treated (w/ random bias)	*+0.018*	*-0.010*	*-0.112 **	*-0.200 **	-	-	0.085
Instruction $\xrightarrow{?}$ Answer	T	F	T	T	F	T	0.066
Implied SCM Type	**II**	**I**	**III**	**III**	**IV**	**III**	-

Table 18: *Identification of causal structures* in tasks running on *Mistral-7B-DPO*.