Teaching Transformers Causal Reasoning through Axiomatic Training

Aniket Vashishtha
UIUC
vashisthaaniket07@gmail.com

Abhinav Kumar
MIT
akumar03@mit.edu

Atharva Pandey
Microsoft Research, India
t-atpandey@microsoft.com

Abbavaram Gowtham Reddy
IIT Hyderabad, India
cs19resch11002@iith.ac.in

Vineeth N Balasubramanian
IIT Hyderabad, India
vineethnb@iith.ac.in

Amit Sharma
Microsoft Research, India
amshar@microsoft.com

Abstract

For text-based AI systems to interact in the real world, causal reasoning is an essential skill. Since interventional data is costly to generate, we study to what extent an agent can learn causal reasoning from passive data. Specifically, we consider an axiomatic training setup where an agent learns from multiple demonstrations of a causal axiom (or rule), rather than incorporating the axiom as an inductive bias or inferring it from data values. A key question is whether the agent would learn to generalize from the axiom demonstrations to new scenarios. For example, if a transformer model is trained on demonstrations of the causal transitivity axiom over small graphs, would it generalize to applying the transitivity axiom over large graphs? Our results, based on a novel axiomatic training scheme, indicate that such generalization is possible. For the transitivity axiom, we find that a 67 million parameter transformer model, when trained on linear causal chains (along with some noisy variations) can generalize well to new kinds of graphs, including longer causal chains, causal chains with reversed order, and graphs with branching; even when it is not explicitly trained for such settings. We extend axiomatic training to a harder task of inferring causation from correlation statements and find similar generalization. On both tasks, our model performs at par (or even better) than many larger language models such as GPT-4, Gemini Pro, and Phi-3. Overall, our axiomatic training framework provides a new paradigm of learning causal reasoning from passive data that can be used to learn arbitrary axioms, as long as sufficient demonstrations can be generated.

1 Introduction

Causal reasoning can be defined as a set of reasoning procedures consistent with pre-defined axioms or rules that are specific to causality [9]. For instance, d-separation and rules of do-calculus can be considered as axioms and specifications of a collider or a backdoor set can be considered as rules that can be derived from axioms. Axioms or rules are incorporated as inductive biases in a machine learning (ML) model, through regularization, model architecture, or the choice of variables for a particular analysis. Depending on the kind of available data—observational, interventional, or counterfactual—Pearl's ladder of causation [5] defines the kinds of causal reasoning that is possible.

As axioms are the building blocks of causality, we study whether it is possible to directly learn the axioms using ML models. That is, rather than learning from data that is the result of axioms obeyed by a data-generating process, what if a model can learn an axiom (and thus causal reasoning) directly from symbolic demonstrations of the axiom? Such a model has the advantage that it can be applied

CaLM: 1st Workshop on Causality and Large Models

for causal reasoning in diverse downstream scenarios, compared to task-specific causal models built using specific data distributions. This question gains relevance as language models make it possible to learn over symbolic data expressed in natural language. In fact, recent studies have evaluated whether large language models (LLMs) can do causal reasoning by creating benchmarks that encode causal reasoning problems in natural language [14, 11, 12].

Specifically, we propose a new way of learning causal reasoning through axiomatic training. We posit that causal axioms can be expressed as the following symbolic tuple, $\langle$*premise*, *hypothesis*, *result*$\rangle$ where *hypothesis* refers to a causal claim and *premise* refers to any relevant information to decide whether the claim is true or not (*conclusion*). The conclusion could simply be "Yes" or "No". For example, one of the conclusions from the collider axiom from [12] can be expressed as: *premise*: "$A \perp\!\!\!\perp B, B \not\perp\!\!\!\perp C, A \not\perp\!\!\!\perp C$"; *hypothesis*: "Does A cause C?"; and the *conclusion* as "Yes". Based on this template, a large number of synthetic tuples can be generated, e.g., by changing the variable names, changing the number of variables, changing the order, and so on. The key question is: if a model is trained on such data, would it learn to apply the axiom to new scenarios?

To answer this question, we train a transformer model from scratch on symbolic demonstrations of the causal irrelevance axiom [9]. To evaluate generalizability, we train on simple chains of the causal irrelevance axiom of size 3-5 nodes and test on multiple different aspects of generalization, including length generalization (chains of size 7-15), name generalization (longer variable names), order generalization (chains with reversed edges or shuffled nodes), and structure generalization (graphs with branching). We find that a model trained on simple chains generalizes to applying the axiom multiple times over larger chains, but it is unable to generalize to the more complex scenarios like order or structure generalization. However, when we train a model on a combined dataset of simple chains and chains with some edges randomly reversed, we find that the model generalizes well across all kinds of evaluation scenarios. Extending the findings on length generalization for NLP tasks [13, 7, 10, 8], we find a critical role of positional embedding in ensuring causal generalization across length and other aspects. Our best model has no positional encoding, although we find that sinusoidal encoding also works well for some scenarios.

The axiomatic training approach also generalizes to a harder problem proposed in [12]. The task is to distinguish correlation from causation given a premise containing statistical independence statements. Solving this task requires knowledge of multiple axioms, including d-separation and Markov property. Using the same method to generate synthetic training data and train the model as above, we find that a transformer trained on task demonstrations over 3-4 variables learns to solve this task for graphs with 5 variables. On this, our model outperforms larger LLMs such as GPT-4, Gemini Pro and Phi-3.

Our work provides a new paradigm of teaching models causal reasoning through symbolic demonstrations of axioms, which we call *axiomatic training*. The data generation and training procedure is general and can be applied to learn any new axiom (including logical axioms), as long as it can be expressed in the symbolic tuple format. More generally, our results contribute to the literature on causal learning from passive data [15] that does not include any active interventions.

2 Learning Causal Axioms In Transformers

Instead of performing causal reasoning using observational or interventional data, we study whether it is possible to learn some of the general rules of causality directly from symbolic axioms. More specifically, we incorporate rules for causal reasoning in transformers as inductive biases. We begin by asking the question "are there any minimal sufficient characterization of causal principles that hold true in general?". There has been a fundamental work from Galles and Pearl [9] where they axiomatize the causal relevance (or equivalently irrelevance). They show that for a given *stable probabilistic* causal model (defined below), there exists a finite set of axioms that are completely characterized by axioms of path interception in corresponding directed graphs. We now study how such causal relevance statements can be incorporated into transformer models.

Let $\mathcal{M} = (\boldsymbol{X}, \boldsymbol{U}, \mathcal{F})$ be a causal model defined over a set of endogenous variables $\boldsymbol{X}$, exogenous variables $\boldsymbol{U}$ and the causal relationship between then defined by set of structural equations $\mathcal{F}$ [9]. Let $\mathcal{G}$ be the causal graph associated with the causal model $\mathcal{M}$ where the nodes $\boldsymbol{V}$ in $\mathcal{G}$ correspond to the variables in $\mathcal{M}$ and an edge $V_i \to V_j$ between any two nodes V_i, V_j denote the causal relationship between them.

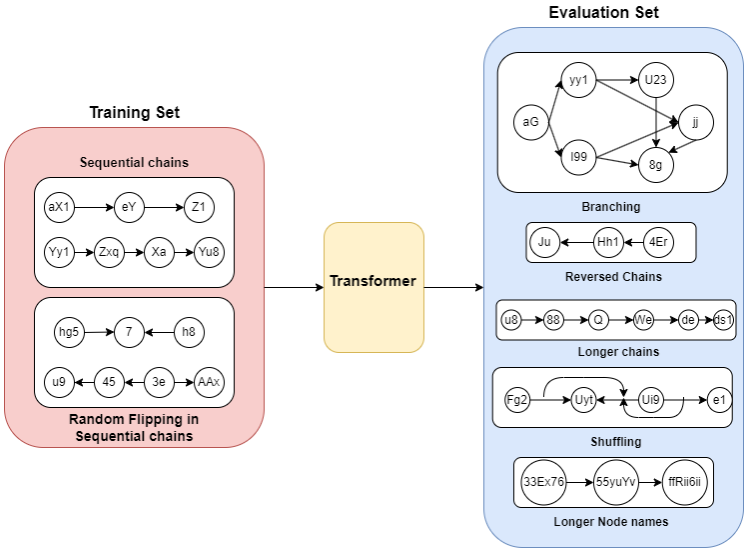

Figure 1: **Evaluating structural generalization of transformers through Axiomatic training**: Our pretraining setup is made of linear sequential chains of small length, no branching, and randomly revrsed edge directions. After training the transformer with our pre-training data D with introduced variability, structural generalization across different dimensions is observed. Specifically across more branched networks with higher average in-degree and out-degree, complete reversals, longer sequences, shuffled natural language statements of sequences and longer node names.

Definition 2.1 (**Causal Irrelevance**, Defn. 7 in [9])**.** *X is probabilistically causally irrelevant to Y given Z, written ($X \nrightarrow Y|Z$) iff: $\mathbb{P}(y|z, do(X) = x) = \mathbb{P}(y|z, do(X) = x')$, $\forall x, x', y, z$ i.e., once we hold Z fixed at z, intervening on X will not change the probability of Y.*

Under the stability assumption (see Assumption G.1), Galles and Pearl [9] characterizes six axioms that completely characterize causal irrelevance (Definition 2.1) or equivalent causal relevance statements after using the corresponding contra-positive statements. An axiom of causal irrelevance is of the form (given in conjunctive normal form):

$$\bigwedge_s \bigvee_t (\boldsymbol{X}_i^{s,t} \nrightarrow \boldsymbol{X}_j^{s,t} | \boldsymbol{X}_k^{s,t}) \implies \bigwedge_l \bigvee_n (\boldsymbol{X}_i^{l,n} \nrightarrow \boldsymbol{X}_j^{l,n} | \boldsymbol{X}_k^{l,n})$$

where $\wedge$ is "logical and", $\vee$ is "logical or" and for a given (s, t) or (l, n) pair, $\boldsymbol{X}_i, \boldsymbol{X}_j, \boldsymbol{X}_k$ are disjoint subsets of observed variables $\boldsymbol{X}$. In the above causal irrelevance statement, if the antecedent is true, the consequent is also true.

Transitivity Axiom: For the scope of our study, we focus on the transitivity axiom (Axiom 3.6, Fig. 7 in [9]) because it is a generic axiom, which can be used to represent complex structures like forks, colliders and chains which are used as building blocks of any causal structure. Below, we restate the transitivity axiom where A, X, Y, Z are endogenous variables of the system. $(\boldsymbol{X} \nrightarrow \boldsymbol{Y}|\boldsymbol{Z}) \implies (\boldsymbol{X} \nrightarrow \boldsymbol{A}|\boldsymbol{Z}) \vee (\boldsymbol{A} \nrightarrow \boldsymbol{Y}|\boldsymbol{Z}) \forall \boldsymbol{A} \notin \{\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}\}$ Which could be equivalency converted into a causal relevance statement by taking the contrapositive: $\exists \boldsymbol{A} \notin \boldsymbol{X} \cup \boldsymbol{Y} \cup \boldsymbol{Z}$ $s.t.$ $\underbrace{(\boldsymbol{X} \to \boldsymbol{A}|\boldsymbol{Z}) \wedge (\boldsymbol{A} \to \boldsymbol{Y}|\boldsymbol{Z})}_{P:\text{premise}} \implies \underbrace{(\boldsymbol{X} \to \boldsymbol{Y}|\boldsymbol{Z})}_{H:hypothesis}$.

2.1 Training data setup for Axiomatic Training

Based on a specific axiom, we can map a hypothesis given the premise to its correct label ('*Yes*' or '*No*'). To create a training dataset, we enumerate all possible tuples of $\{(P, H, L)\}_N$ where P is the premise, H is the hypothesis and L is the label *(Yes/No)* for a particular setting of the variables X, Y, Z, A. Given a premise P based on a given causal graph, if the hypothesis can be derived by applying the specified axiom (once or multiple times), then label L is *Yes*; otherwise, *No*. For example, suppose the underlying true causal graph of a system has the topology of a chain, $X_1 \to X_2 \to X_3 \to \cdots \to X_n$. Then, a possible premise could be $X_1 \to X_2 \wedge X_2 \to X_3$, and the corresponding hypothesis $X_1 \to X_3$ will have label *Yes* whereas another hypothesis $X_3 \to X_1$

will have label *No*. The above axiom could be inductively applied multiple times to generate more complex training tuples.

For our training setup, a synthetic dataset D is constructed with N axiomatic instances generated using the transitivity axiom. Each instance in D is structured in the form of a premise P, which is the natural language expression of a causal structure (e.g., "X causes Y. Y causes Z"), followed by the hypothesis in the form of a question H_q (e.g., "Does X cause Y?"), which is then followed by the final label L (e.g., "Yes" or "No"). Each instance in D is structured as (P_i, H_{ij}, L_{ij}); $j \in \{1, \ldots, \binom{n}{2}\}$where n is the number of nodes in each ith premise, thus effectively covering all pairs of nodes in each unique chain of a given causal graph.

2.2 Data Perturbation: A Key to Model Generalization

Variability or diversity in training data in the form of perturbation helps aid model generalization [16]. For axiomatic training, we provide structured perturbation to extend our transformer model across complex structures for which it was not explicitly trained on. We introduce perturbations at multiple levels in the training data to maximize diversity in the training set distribution, as explained below.

1. **Node names**: Each node in the transitivity chain is represented by an alphanumeric name comprising 1-3 characters. The length of a name and the specific characters are randomly selected during data generation.
2. **Causal Graph Topology**: We consider two main types of causal graphs for the training set.
   (a) **Sequential**: All causal edges are directed forward, thus forming a typical transitivity chain, e.g. $X \rightarrow Y \rightarrow Z$.
   (b) **Random Flipping**: Given a chain of sequential nodes, we randomly reverse some edges creating complexity by disrupting direct paths between subsequent nodes (eg. $X \rightarrow Y \leftarrow Z$). This can be expressed simply through natural language like: *"X causes Y. Z causes Y."*
3. **Length level**: To facilitate transformers understanding of the axiom, we incorporate chains of varying lengths, ranging from 3 to 6 nodes in our training set.

Random flipping introduces forks and colliders, which form the building blocks of any causal DAG.

2.3 Assessing Axiomatic Learning in Transformers

While we restrict training to simple chain-like sequences, we consider a much more complex evaluation set, to avoid any conclusions based on model overfitting (e.g., learning shortcuts or correlation-based features). To tackle this, we evaluate our model across multiple types of complex structures that are not seen during training. We divide the structural complexities as follows:

1. **Length**: Evaluating whether our model accurately infers causal relationships for sequences or chains (both sequential and ones with random flipping) longer than those in the train set.
2. **Node Name Shift**: Testing the model's performance with longer node names, increasing from 1-3 characters used in the training set to 8-10 characters. [12] found that shifts in node names during language model fine-tuning lead to generalization failures in distinguishing causal from correlational relationships, despite strong in-distribution performance.
3. **Order of Chains**: a) **Completely reversed chains**: This evaluation is inspired by the reversal curse [6] that revealed generalization failure of LLMs in answering questions in reversed sequences despite knowing the answers in the original order. We evaluate the capacity of axiomatic training to enable reasoning over reversed chains even when not explicitly trained on any completely reversed chains. A completely reversed chain will be of the form $X \leftarrow Y \leftarrow Z$ with its natural language representation as: *"Y causes X. Z causes Y."*, where X, Y, Z are replaced by random alphanumeric names. b) **Shuffling of Sequences:** Causal sequences with random edge flips, as defined in 2.2 represented by natural language statements sequentially *(A causes B. B causes C ...)*, are shuffled to add complexity and break sequential order. This tests transformers' ability to infer accurate relationships regardless of sequence order of premise.
4. **Branching**: Causal graphs with dense branching pose a challenging evaluation task. While the training set comprises simplistic linear sequences, this evaluation setup involves multiple branches, colliders, forks, and chains in one network, with significantly high complexity. We measure complexity of a graph using its branching factor: Number of edges/Number of nodes.

Unlike length and node name generalization, the reversal and branching evaluation setups change the causal structure and hence better evaluate whether the model has learnt accurate representations for causal structure. Branching is perhaps the most challenging since it contains new structures (due to a higher branching factor) that were unseen during training.

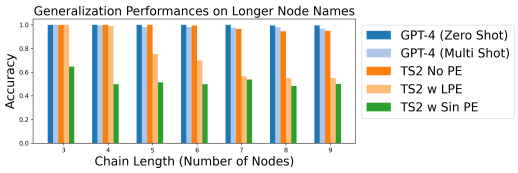

Figure 2: Evaluating generalization on causal sequences (without random flipping) with longer node names (than the ones used in sequences in train set). TS-2 training set with no positional encoding leads to the best performance.

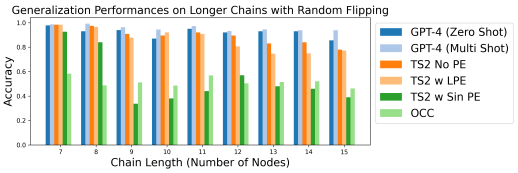

Figure 3: Generalizing to longer unseen causal sequences (>6 nodes) with random flipping on TS2 and OCC (with NoPE) train sets. OCC-trained models struggle due to limited edge-level variability, while TS2 NoPE consistently performs well.

3 Application 1: Learning the Causal Transitivity Axiom

Training Data: Our training set comprises approximately 175K instances of sequential chains, ranging from 3 to 6 nodes in size. We employ three training data versions. **1) Only causal chains (OCC).** Sequential chains (175K) without any random flip of edges; **2) Training Setup 1 (TS1).** Combines causal chains (101K) and sequences with randomly flipped edges (73K), while ensuring that (reversals removed and model re-trained for evaluating on reversal chains); **3) Training Setup 2 (TS2).** Combines causal chains (132K) and sequences with flipped edges (42K) with a higher fraction of causal chains. We exclude complete reversals in TS1 to check generalization of the model.

Architectural and Model Training Details: We train a GPT2 [17] decoder based 67 million parameter model from scratch on our transitivity based dataset. Our model is trained for 100 epochs (due to optimal loss convergence), with 1e-4 learning rate. Our GPT2 based model using AdamW optimizer has 12 attention layers, 8 attention heads and 512 embedding dimensions. We evaluate the impact of different Positional Encodings on the generalization capabilities of the transformer model. We evaluate how well models perform without positional encodings (or NoPE) [13], with Sinusoidal PEs (SPE) [20] and Learnable PEs (LPE) [17]. Past work has shown how removing PEs lead to better generalization performance for decoder models, which pick up an implicit positional understanding due to the nature of next token prediction based training of language models [13]. We further evaluate how well NoPE aids the model's generalization for our setting. Details of our custom tokenizer and LLM baselines are in Appx. § D and § E, respectively.

Loss Function: We optimize loss based on the ground truth label for all settings, represented as $\mathop{\mathbb{E}}_{P,H_q,L\sim P_{\text{train}}} -\log(P(L|P,H_q))$. Our earlier analysis indicated promising results with this approach compared to using next token prediction loss.

Results - Data diversity matters: Models with No PEs generalize well to longer lengths, even though they are only trained on chain length of 3-6. Model trained on only sequential chain (OCC), however, only generalize to longer Sequential chains (Table 5) but not to other DAG structures (Figure 3 for edge flip, Figure 5 for reversal, Table 4 for branching). Models trained on TS1 or TS2 generalize across all scenarios, including edge flip, order, and branching. As sequence length increases without random flipping, TS2 performs best, likely due to less noise in train set from fewer flipped sequences. This suggests that while variability aids structural generalization, excessive variability can hinder it. Reversal and shuffling evaluations are challenging as the model hasn't been trained to learn causal structures independent of order. Despite this, our model achieves 95% accuracy on reversal tasks, outperforming much larger models like Gemini Pro and Phi-3. A similar pattern is observed for shuffling, where `TS2 (NoPE)` significantly surpasses larger baseline models with ease. Branching is challenging due to increased inter-node connections, since training set only contains linear chains, but despite that our models achieve good performance with a margin of 20% over random baselines. Our findings underscore the significance of diverse data for generalization.

Axiom-trained transformer generalizes to complex causal scenarios: The model trained with TS2 performs well across all setups. Even though our model is not explicitly trained on completely reversed chains, it still performs at par with GPT-4 (Fig. 5). Models trained on TS1 and TS2, trained explicitly without shuffling, show similar trends when evaluated on shuffled sequences with random flipping (Tab. 3). However, transformer trained on OCC setup fails for such settings. Our best models

(NoPE trained on TS1 and TS2) outperforms random baselines (50%) and other billion scale models like Gemini Pro and Phi-3 (and GPT-4 in multiple cases) on both zero shot and multi shot settings.

Role of positional encodings: We also study the effect of positional encoding. Sinusoidal (SPE) and Learnable PEs (LPE) perform well on longer chains but poorly when node names' length increases, even with small chain lengths (Figure 2). Similarly, SPE does not perform well across different structural dimensions like branching, and order based settings. NoPE performs consistently well across all settings, indicating its generalization power even beyond length. See App. H for details.

4 Application 2: Infer causation from correlation statements (Corr2Cause)

The study extends the evaluation of transformers' generalization from simple transitivity in small causal chains to more complex causal tasks. Specifically, it applies axiomatic training to inferring causation from correlational statements in observational data involving graphs with 3 to 6 nodes. The goal is to determine the truth value of hypotheses (e.g., Parent, Ancestor, Collider) regarding direct or indirect relationships between nodes. This task is more challenging as it requires understanding d-separation, the Markov property, and mapping correlational statements to multiple causal graphs within the Markov Equivalence Class. We train the same architecture from our transitivity experiments for 100 epochs using NoPE on a dataset with 3-4 node graphs, testing it on 5-node correlational statements due to its strong OOD performance.

To aid generalization, we take inspiration from our transitivity-based experiments and create different combinations of randomly created alphanumeric node names. We then derive a training set from the original dataset by instantiating the correlational statements with different combinations of alphanumeric node names. We balance the dataset by sampling equally from both classes to avoid bias in our transformer model to get a train set with 113099 instances. Then, we create a test set with 1000 randomly sampled instances of correlational statements for 5-node graph networks.Given the complexity of each input description, we tokenize the input text at the word level instead of character level and use the same node names for evaluation as in the training set to avoid potential out-of-vocabulary issues.

Model	Precision	Recall	F1 Score	Accuracy
Ours	**0.72**	0.50	0.59	**0.64**
Zero-Shot				
Phi-3	0.52	0.60	0.56	0.52
Gemini pro	0.52	0.59	0.55	0.52
GPT-4	0.59	0.50	0.54	0.58
Multi-Shot				
Phi-3	0.57	0.67	0.61	0.58
Gemini pro	0.51	**0.74**	0.60	0.52
GPT-4	0.66	0.56	**0.61**	**0.64**

Table 1: Correlation to Causation Experiments adapted from [12]

Comparison with Baselines: As reported in [12], due to the complexity of the task, we find that pre-trained LMs such as Gemini Pro and Phi-3 perform similar to a random guess (52% accuracy). While GPT-4 does perform slightly better, it's performance is still low (58% accuracy). Remarkably, our small transformer model performs better than all baselines with 64% accuracy; 6% points higher than GPT-4. With further exploration of different training setups, axiomatically-trained transformer models may be optimised further for such causal reasoning tasks.

5 Discussion and Conclusion

We propose an axiomatic training method to teach causal reasoning to transformers. Our results show that a transformer can learn to apply a causal axiom and generalize to multiple, complex graph structures that were not seen during training. Future work includes extending axiomatic training to learn multiple axioms, use naturally-occuring text data, and explore other training losses.

Applicability to Causal Tasks: While our current work focuses on the transitivity axiom for causal relevance, extending the work to other causal axioms from [9] is an interesting research direction. In addition, we may consider other axioms that are relevant for downstream tasks such as effect inference. For example, if a transformer model can be trained to validate the d-separation rule—given two variables X and Y, are they independent given a variable set Z?—then repeated applications of the rule can be used to derive a backdoor set.

Generalization to Logical Reasoning: While we focused on causal reasoning, axiomatic training can be applied to any formal system based on axioms. For instance, the same axiomatic training procedure can be used for teaching LMs logical reasoning tasks such as deductive reasoning [18].

References

[1] Phi-3 technical report: A highly capable language model locally on your phone, 2024.

[2] Gemini: A family of highly capable multimodal models, 2024.

[3] Gpt-4 technical report, 2024.

[4] Taiyu Ban, Lyvzhou Chen, Xiangyu Wang, and Huanhuan Chen. From query tools to causal architects: Harnessing large language models for advanced causal discovery from data. *arXiv preprint arXiv:2306.16902*, 2023.

[5] Elias Bareinboim, Juan Correa, Duligur Ibeling, and Thomas Icard. On pearl's hierarchy and the foundations of causal inference (1st edition). In Hector Geffner, Rita Dechter, and Joseph Halpern, editors, *Probabilistic and Causal Inference: the Works of Judea Pearl*, pages 507–556. ACM Books, 2022.

[6] Lukas Berglund, Meg Tong, Max Kaufmann, Mikita Balesni, Asa Cooper Stickland, Tomasz Korbak, and Owain Evans. The reversal curse: Llms trained on "a is b" fail to learn "b is a", 2024.

[7] Satwik Bhattamishra, Kabir Ahuja, and Navin Goyal. On the ability and limitations of transformers to recognize formal languages, 2020.

[8] Daniel Furrer, Marc van Zee, Nathan Scales, and Nathanael Schärli. Compositional generalization in semantic parsing: Pre-training vs. specialized architectures, 2021.

[9] David Galles and Judea Pearl. Axioms of causal relevance. *Artificial Intelligence*, 97(1):9–43, 1997. ISSN 0004-3702. doi: https://doi.org/10.1016/S0004-3702(97)00047-7. URL https://www.sciencedirect.com/science/article/pii/S0004370297000477. Relevance.

[10] Adi Haviv, Ori Ram, Ofir Press, Peter Izsak, and Omer Levy. Transformer language models without positional encodings still learn positional information. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang, editors, *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 1382–1390, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.findings-emnlp.99. URL https://aclanthology.org/2022.findings-emnlp.99.

[11] 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. Cladder: Assessing causal reasoning in language models, 2024.

[12] Zhijing Jin, Jiarui Liu, Zhiheng Lyu, Spencer Poff, Mrinmaya Sachan, Rada Mihalcea, Mona Diab, and Bernhard Schölkopf. Can large language models infer causation from correlation?, 2024.

[13] Amirhossein Kazemnejad, Inkit Padhi, Karthikeyan Natesan Ramamurthy, Payel Das, and Siva Reddy. The impact of positional encoding on length generalization in transformers. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 24892–24928. Curran Associates, Inc., 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/4e85362c02172c0c6567ce593122d31c-Paper-Conference.pdf.

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

[15] Andrew Lampinen, Stephanie Chan, Ishita Dasgupta, Andrew Nam, and Jane Wang. Passive learning of active causal strategies in agents and language models. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 1283–1297. Curran Associates, Inc., 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/045c87def0c02e3ad0d3d849766d7f1e-Paper-Conference.pdf.

[16] Bai Li, Changyou Chen, Wenlin Wang, and Lawrence Carin. Certified adversarial robustness with additive noise, 2019.

[17] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2018. URL https://d4mucfpksywv.cloudfront.net/better-language-models/language-models.pdf.

[18] Abulhair Saparov, Richard Yuanzhe Pang, Vishakh Padmakumar, Nitish Joshi, Seyed Mehran Kazemi, Najoung Kim, and He He. Testing the general deductive reasoning capacity of large language models using ood examples, 2023.

[19] Aniket Vashishtha, Abbavaram Gowtham Reddy, Abhinav Kumar, Saketh Bachu, Vineeth N Balasubramanian, and Amit Sharma. Causal inference using llm-guided discovery, 2023.

[20] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2023.

Appendix

A Performance Results for Different Evaluation Setups

Tables 2 and 3 shows the results of generalization to reversal and shuffling; Table 5 shows the results on length generalization; and Table 4 shows the results on branching generalization. Figures 5 and **??** highlight generalization performance on reversal and longer chains.

Model	3	4	5	6
Baselines				
Zero Shot				
GPT-4	0.97	0.99	0.98	0.92
Gemini Pro	0.61	0.59	0.66	0.62
Phi-3	0.80	0.69	0.73	0.69
Multi Shot				
GPT-4	**1.00**	**1.00**	**1.00**	**0.99**
Gemini Pro	0.95	0.87	0.77	0.71
Phi-3	0.93	0.89	0.75	0.75
Axiomatic Training				
TS1 w NoPE	0.98	0.99	0.92	0.91
TS1 w SPE	**1.00**	0.99	0.99	0.97
TS2 w NoPE	0.99	0.99	0.95	0.94
TS2 w SPE	0.98	0.97	0.93	0.94
TS2 w LPE	0.99	0.98	0.95	0.97
OCC w NoPE	0.33	0.18	0.10	0.09

Table 2: Evaluated on completely reversed chains, even when not explicitly trained on reversed chains. Model trained only on sequential chains (OCC), performs the worst, while transformer trained on both Sequential chains, and sequences with random flipping perform the best (training sets: TS1 and TS2). Accuracy Metric reported. This setup is inspired by the [6] setup.

Model Config	3	4	5	6	7	8	9
Baselines							
Zero Shot							
GPT-4	0.99	0.97	0.89	0.85	**0.95**	**0.90**	0.90
Gemini Pro	0.75	0.73	0.72	0.76	0.71	0.68	0.74
Phi-3	0.88	0.86	0.82	0.79	0.76	0.73	0.79
Multi Shot							
GPT-4	**1.00**	**0.99**	**0.97**	**0.95**	0.94	**0.90**	**0.92**
Gemini Pro	0.95	0.85	0.83	0.79	0.79	0.73	0.75
Phi-3	0.88	0.83	0.82	0.80	0.83	0.76	0.78
Axiomatic Training							
TS1 NoPE	**1.00**	0.94	0.87	0.84	0.80	0.76	0.73
TS1 LPE	**1.00**	0.95	0.87	0.83	0.78	0.78	0.71
TS1 SPE	**1.00**	0.94	0.86	0.83	0.76	0.73	0.68
TS2 NoPE	**1.00**	0.95	0.87	0.84	0.79	0.76	0.73
TS2 w LPE	**1.00**	0.94	0.87	0.84	0.80	0.76	0.73
TS2 w SPE	0.99	0.94	0.89	0.84	0.75	0.74	0.49
OCC w NoPE	0.69	0.62	0.57	0.54	0.57	0.53	0.52

Table 3: Evaluated on shuffled natural language sequence of randomly flipped sequence. Random flipping, length (7-9) and random flipping add complexity to the evaluation setup, since our model is not trained on shuffled set. Accuracy metric is reported

Model	5		8		10		12	
	BF=2	BF=1.4	BF=2	BF=1.4	BF=2	BF=1.4	BF=2	BF=1.4
Baselines								
Zero shot								
GPT-4	0.98	0.95	0.91	0.90	0.84	0.88	0.82	0.86
Gemini Pro	0.77	0.74	0.72	0.76	0.71	0.73	0.73	0.71
Phi-3	0.87	0.83	0.82	0.79	0.77	0.77	0.75	0.80
Multi shot								
GPT-4	**0.99**	**0.97**	**0.94**	**0.93**	**0.90**	**0.94**	**0.89**	**0.93**
Gemini Pro	0.81	0.76	0.77	0.79	0.75	0.77	0.78	0.79
Phi-3	0.77	0.78	0.79	0.82	0.78	0.794	0.80	0.79
Axiomatic Training								
OCC w NoPE	0.52	0.51	0.53	0.52	0.52	0.55	0.49	0.47
TS1 w LPE	0.79	0.84	0.71	0.76	0.68	0.69	0.65	0.65
TS1 w SPE	0.72	0.79	0.63	0.64	0.56	0.61	0.52	0.59
TS1 w NoPE	0.77	0.84	0.73	0.76	0.68	0.70	0.62	0.66
TS2 w LPE	0.72	0.80	0.61	0.71	0.62	0.63	0.56	0.63
TS2 w SPE	0.52	0.70	0.49	0.49	0.49	0.49	0.51	0.52
TS2 w NoPE	0.83	0.86	0.74	0.77	0.69	0.74	0.64	0.70

Table 4: Evaluated on branched graphs created using Erdos Renyl, with varying branching factors (calculated by number of edges/number of nodes). TS1 and TS2 denote Pretraining data setup 1 and 2 from Section 3. OCC setup denotes Only sequential Causal Chains with no random flipping. SPE: Sinusoidal PE, LPE: Learnable PE, w/o PE: No PE. Decoder model remains the same across all setups (67 Million parameter), Accuracy metric is used

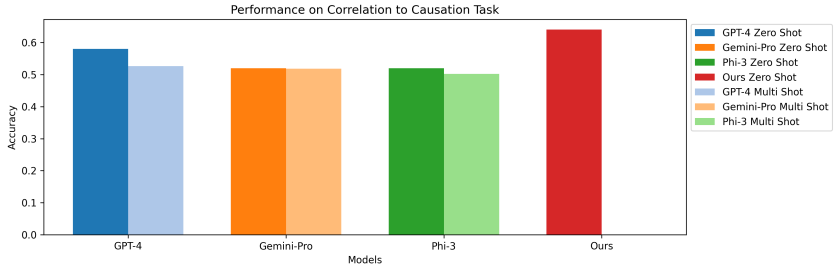

Figure 4: Correlation to Causation Experiments adapted from [12]. Axiomatic training setup aids generalization even for complex causal tasks, while bigger LLMs struggle on the same in zero-shot setting. Refer section 4 for details regarding experimental setup and result trends.

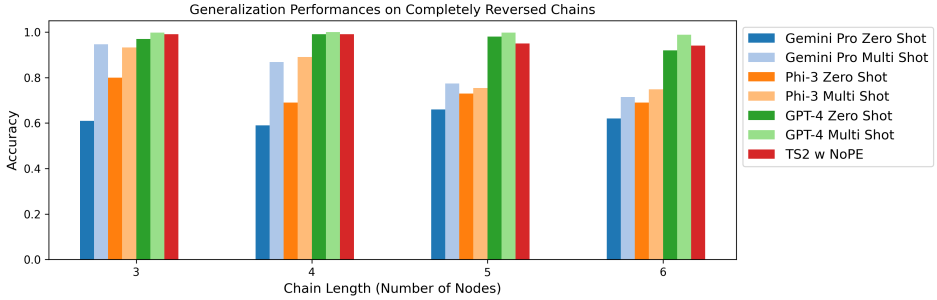

Figure 5: Performance comparison of our best performing transformer model trained on TS2 with NoPE (trained without any completely reversed chains), against larger models like GPT-4, Gemini Pro and Phi-3.

Model	7		8		9		10		11		12		13		14		15	
	FS	RF	FS	RF	FS	RF	FS	RF	FS	RF	FS	RF	FS	RF	FS	RF	FS	RF
Baselines																		
Single Shot																		
GPT-4	0.95	0.98	0.97	0.93	0.87	0.94	0.91	0.87	**0.90**	0.95	0.92	0.92	0.85	0.93	**0.93**	0.93	**0.89**	0.86
Gem-Pro	0.63	0.73	0.69	0.74	0.64	0.75	0.65	0.81	0.72	0.78	0.60	0.80	0.59	0.68	0.67	0.64	0.61	0.66
Phi-3	0.81	0.85	0.96	0.85	0.85	0.85	0.87	0.89	**0.90**	0.86	0.84	0.85	0.91	0.84	0.90	0.80	0.78	0.85
Multi Shot																		
GPT-4	0.97	**0.99**	0.93	**0.99**	**0.92**	**0.96**	0.88	**0.94**	0.89	**0.97**	0.89	**0.93**	0.88	**0.95**	**0.93**	**0.94**	0.86	**0.94**
Gem-Pro	0.80	0.82	0.81	0.79	0.78	0.81	0.67	0.79	0.73	0.82	0.74	0.83	0.67	0.78	0.72	0.78	0.68	0.78
Phi-3	0.83	0.92	0.89	0.88	0.75	0.86	0.66	0.87	0.80	0.90	0.80	0.85	0.79	0.82	0.71	0.81	0.72	0.82
Axiomatic Training																		
TS1 w NoPE	**1.00**	**0.99**	0.95	0.96	0.88	0.89	0.76	0.88	0.73	0.90	0.77	0.92	0.61	0.82	0.67	0.78	0.68	0.81
TS1 w LPE	0.98	0.96	0.92	0.97	0.77	0.90	0.59	0.87	0.57	0.86	0.57	0.84	0.55	0.73	0.51	0.76	0.50	0.68
TS1 w SPE	0.99	0.95	0.95	0.94	0.86	0.76	0.80	0.75	0.76	0.79	0.84	0.68	0.79	0.63	0.85	0.65	0.77	0.69
TS2 w NoPE	**1.00**	0.98	**0.99**	0.97	**0.92**	0.91	0.88	0.90	0.86	0.92	**0.95**	0.90	**0.96**	0.83	0.81	0.84	0.85	0.78
TS2 w LPE	**1.00**	0.98	0.88	0.97	0.80	0.88	0.62	0.92	0.66	0.91	0.64	0.81	0.65	0.75	0.62	0.75	0.62	0.77
TS2 w SPE	0.95	0.93	0.81	0.84	0.56	0.34	0.50	0.38	0.50	0.44	0.51	0.57	0.46	0.74	0.52	0.75	0.50	0.77
OCC w NoPE	0.98	0.58	0.79	0.49	0.86	0.51	**0.92**	0.49	0.72	0.57	0.90	0.50	0.81	0.52	0.84	0.52	0.83	0.46

Table 5: Results on longer chains of linear sequential chains with all edges in forward direction (Only causal chains or forward sequence denoted using FS) and sequences with randomly flipped edges (Random flipping so denoted with RF). TS1 and TS2 denote Pretraining data setup 1 and 2 from Section 4. SPE: Sinusoidal PE, LPE: Learnable PE, w/o PE: No PE. Model remains the same across all setups (67 Million parameter based). For longer chains, NoPE performs best on sequential linear setup. Accuracy metric is used

B Example of Instances from Corr2Causation Benchmark

Following is one of the example instances from the benchmark of Corr2Cause [12], where the model has to infer causal relationships from correlational statements.
Premise*: Suppose there is a closed system of 4 variables, A, B, C and D. All the statistical relations among these 4 variables are as follows: A correlates with B. A correlates with C. A correlates with D. B correlates with C. B correlates with D. C correlates with D. However, B and D are independent given A. B and D are independent given A and C. C and D are independent given A. C and D are independent given A and B. Hypothesis: There exists at least one collider (i.e., common effect) of A and B.*

C Example of Instances from Our Evaluation Sets

Following is one of the instances from the evaluation set for sequences with random flipping, where the model has to infer causal relationships from natural language statements.
Premise*:V causes f. f causes jbj. ag causes jbj. ag causes rBz. rBz causes Tm2. EaT causes Tm2. Hypothesis: Does V cause f?*
Following is one of the instances from the evaluation set for sequences with reversals, where the model has to infer causal relationships from natural language statements.
Premise*:LQw causes e2. p causes LQw. u causes p. a causes u. Hypothesis: Does e2 cause LQw?*

D Custom Tokenizer details

For tokenization, we develop a custom tokenizer. Alphanumeric node names are tokenized at a character level, while terms like 'causes', 'Does', 'cause', 'Yes', and 'No' are tokenized at the word level. The intuition behind this approach is to avoid out of vocabulary (OOV) tokens in the test time, since the alphanumeric node names of test set are different then the training set and are created randomly, therefore creating a high chance of coming across unseen node names. Following this approach, the vocab size of our transformer model is extremely constrained (69) since it only contains 4-5 word tokens and rest alphanumeric characters along with punctuation marks.

E Baselines: How well do LLMs do on these evaluations?

Given recent work on how LLMs can be leveraged for causal reasoning [14, 19, 4], we include language models such as GPT-4 *(gpt-4-32k)* [3], Gemini *(gemini-pro)* [2] and Phi-3 *(Phi-3-mini-128k-instruct)* [1] as baselines. Note that each of these models is significantly larger than our model and known to perform well on reasoning tasks, with the smallest baseline model Phi-3 having 3.8 billion parameters. We incorporate both commercial (GPT-4 and Gemini Pro) and open-source (Phi-3) models covering a range of size and capabilities. To evaluate the baseline models, we follow a simple zero-shot prompting strategy. For each tuple, we provide the natural language expression of the causal graph *(Premise)* followed by the question *(Hypothesis)* and prompt the LM to answer it in either 'Yes' or 'No' *(Label)*. Here is an example prompt: *"EX causes T. T causes 9. 9 causes W. W causes 7. 7 causes M. M causes a. Does EX cause T? Answer in 'Yes' or 'No' only."*

E.1 Multi-Shot Prompt

To evaluate the performance of baseline LLMs when given in-context examples of similar problems, we present few-shot instances from our training data that include sequential causal chains, along with a few examples with random flipping of edges. We ensured that all multi-shot instances were sourced exclusively from the training set, with no examples from the evaluation set provided to the LLM

E.1.1 Cause-Effect Inference Task

Chain lengths of the in context examples ranged from 3 to 6 to maintains consistency with the training and testing paradigm used for our 67-million-parameter model.

The following multi-shot prompt was used to evaluate the baselines and models across different test sets, assessing their generalization based on length, order, and branching.

Following the given examples answer the question regarding causal relationship between two variables: '5e0 causes vAf. vAf causes VO. Does vAf cause VO?: Yes'
'5e0 causes vAf. vAf causes VO. Does vAf cause 5e0?: No'
'e0F causes Z. Z causes 0U. 0U causes mR. mR causes 1L. Does mR cause 1L?: Yes'
'e0F causes Z. Z causes 0U. 0U causes mR. mR causes 1L. Does Z cause e0F?: No'
'b causes K. K causes qPv. 5 causes qPv. Does b cause qPv?: Yes'
'b causes K. K causes qPv. 5 causes qPv. Does b cause 5?: No'
'Mhb causes t0a. 6Eh causes Mhb. NS causes 6Eh. n causes NS. n causes xu. Does xu cause 6Eh?: No'
'Mhb causes t0a. 6Eh causes Mhb. NS causes 6Eh. n causes NS. n causes xu. Does n cause NS?: Yes'

E.2 Corr2Cause

Below is the MultiShot Prompt for the Corr2Cause Experiment.
Premise: Suppose there is a closed system of 4 variables, R, sG, vE and Y. All the statistical relations among these 4 variables are as follows: R correlates with vE. R correlates with Y. sG correlates with vE. sG correlates with Y. vE correlates with Y. However, R is independent of sG. Hypothesis: There exists at least one confounder (i.e., common cause) of vE and Y. : YES.
Premise: Suppose there is a closed system of 4 variables, uV, S, v and pPf. All the statistical relations among these 4 variables are as follows: uV correlates with v. uV correlates with pPf. S correlates with v. S correlates with pPf. v correlates with pPf. However, uV is independent of S. Hypothesis: There exists at least one confounder (i.e., common cause) of uV and v. : NO.
Premise: Suppose there is a closed system of 3 variables, 39, 52 and fM. All the statistical relations among these 3 variables are as follows: 39 correlates with C. 52 correlates with fM. However, 39 is independent of 52. Hypothesis: There exists at least one collider (i.e., common effect) of 39 and 52. : YES
Premise: Suppose there is a closed system of 3 variables, mFv, lth and HVD. All the statistical relations among these 3 variables are as follows: mFv correlates with HVD. lth correlates with HVD. However, mFv is independent of lth. Hypothesis: There exists at least one collider (i.e., common effect) of lth and HVD. : NO
Premise: Suppose there is a closed system of 4 variables, g1L, wlA, oO and D. All the statistical relations among these 4 variables are as follows: g1L correlates with oO. g1L correlates with Z9. wlA correlates with oO. wlA correlates with Z9. oO correlates with Z9. However, g1L is independent of wlA. wlA and Z9 are independent given g1L and oO. Hypothesis: wlA is a cause for Z9, but not a direct one. : YES.
Premise: Suppose there is a closed system of 4 variables, 6na, lWS, rw and IG. All the statistical relations among these 4 variables are as follows: 6na correlates with rw. 6na correlates with IG. lWS correlates with rw. lWS correlates with IG. rw correlates with IG. However, 6na is independent of lWS. 6na and IG are independent given lWS and rw. 6na and IG are independent given rw. lWS and IG are independent given 6na and rw. lWS and IG are independent given rw. Hypothesis: rw is a cause for lWS, but not a direct one. : NO.
Premise: Suppose there is a closed system of 3 variables, VR4, zf and D. All the statistical relations among these 3 variables are as follows: VR4 correlates with D. zf correlates with D. However, VR4 is independent of zf. Hypothesis: zf directly causes D. : YES.
Premise: Suppose there is a closed system of 3 variables, uj, x and rW. All the statistical relations among these 3 variables are as follows: uj correlates with rW. x correlates with rW. However, uj is independent of x. Hypothesis: uj directly causes x. : NO.

F Compute Resources

We run our experiments on 1 A-100 GPU system, for training our models from scratch and evaluating them. We use 1 GPT-4 API for baseline experiments, while Phi-3 and Gemini Pro provide free resources for model inference.

G Formal Definitions of Axioms of Causal Irrelevance

Here we restate the stability assumption for a causal model from [9] that gives a richer set of finite axiomatization for probabilistic causal irrelevance.

Assumption G.1 (Stability, Definition 9 in [9])**.** *Let $\mathcal{M}$ be a causal model. Then an irrelevance $(X \nrightarrow Y|Z)$ in $\mathcal{M}$ is stable it is shared by all possible probability distribution over $\mathcal{M}$. The causal model $\mathcal{M}$ is stable if all of the irrelevances in $\mathcal{M}$ are stable.*

H Trend Breakdown of Results

H.1 Generalization to Complex Causal Scenarios

We present results on how well an axiomatically trained transformer can generalize to larger and more complex causal graphs, and how it compares to pre-trained LLMs in both zero-shot and multi-shot settings.

Length Generalization: Table 5 shows accuracy of different models when evaluated on longer causal sequences that were not seen during training. Among the baseline pre-trained LMs, GPT-4 obtains the highest accuracy on both standard and randomly flipped chains for the few shot setting, where some examples instances were shown in the model prompt for sequences with length ranging from 3 to 6. It is remarkable that our TS2 (NoPE) model obtains competitive performance to the trillion-scale GPT-4 model, even though it had never seen larger sequences (length 6) during training. Specifically, for chains of size 7-12, TS2 (NoPE) achieves higher or comparable accuracy to GPT-4 in the multi-shot setting. Similar trends are observed for chains of size 7-13 when compared to GPT-4 in the zero-shot setting, both for standard and randomly flipped chains. Its accuracy decreases for chains of length 14-15 (0.85 for standard chains and 0.78 for randomly flipped chains) but is still significantly higher than that of LMs like Gemini-Pro and Phi-3. Although in-context examples improve the performance of baseline LLMs, TS2 (NoPE) still easily outperforms both Gemini Pro and Phi-3 in the multi-shot setting. Note that a random prediction would yield a 50% accuracy, indicating that the axiomatically-trained TS2 (NoPE) model can generalize its reasoning to causal chains much longer than 6 even though it was trained only on chains *upto* length 6.

Node Name Shift: For models trained on TS2 dataset, we also evaluate generalization to changes in variable names (Figure 2). We find that TS2 (NoPE) is robust to node name changes and retains its high accuracy as new, longer names are introduced. It also retains its generalizability to longer sequences with new node names, performing similarly to GPT-4.

Order of Causal Sequences: We now consider how variations in the causal structure impact generalization of axiomatically-trained models. In Table 3, we consider the complex evaluation setup $\mathbf{MultiEval_{SLR}}$ that includes shuffled order of causal sequences with random flipping for increasing length (even beyond the ones in train set). On this task, TS2 (NoPE) obtains higher accuracy than Gemini Pro and Phi-3 on chains of length up to 8. At length 9, TS2 (NoPE) obtains 0.73 accuracy which is comparable to Gemini Pro (0.74) and significantly better than random baseline. Typically there is a huge performance improvement for baseline LLMs like Gemini pro (upto 20%), but still TS2 (NoPE) and TS1 (NoPE) easily perform better across majority of the sequence lengths. While the multi-shot setting generally improves performance compared to the zero-shot setting, in some cases, the performance remains unchanged. TS2 (NoPE) performs better than both Gemini-Pro and Phi-3 across both zero and multi shot settings.

We observe a similar pattern for evaluation on completely reversed sequences in Table 2.

This is an extreme case of out-of-distribution data since most causal edges are left-to-right in the training data whereas the test data contains all right-to-left edges. On this task, our axiomatically trained model TS2 (NoPE) outperforms GPT-4 (zero shot) when restricted to chain lengths of 3-6. In particular, its accuracy (0.94 for chains of length 6) is substantially higher than Gemini Pro and Phi-3 (0.62 and 0.69 respectively for zero shot setting). While multi-shot prompting leads to improved performance of baseline LLMs over zero shot prompting by a margin of 20% (for 5-6 chain lengths), TS2 (NoPE) is still impressively able to outperform bigger models (Gemini Pro and Phi-3) across both settings.

Branching: Finally, we consider the hardest evaluation task involving non-linear chains where we introduce general Erdos-Renyi graphs as the causal sequences while the training data contains only linear chains. Here the length of sequence corresponds to the number of nodes in the graph and we study the performance differences as the branching factor is varied. While GPT-4 obtains the best accuracy (multi shot) across increasing graph sizes, our `TS2 (NoPE)` model obtains higher accuracy than Gemini Pro (zero-shot) for all graph sizes except one (for 1.4 branching factor). Even when evaluated on graphs with 12 nodes and 1.4 branching factor, the `TS2 (NoPE)` model obtains 70% accuracy, significantly better than random baseline (50%). Note that the training data only included graphs with a branching factor of 1.

Summary: Across all evaluation setups, our axiomatically trained model `TS2 (NoPE)` performs significantly better than random baselines even as chain lengths are increased beyond its training data. In particular, even though our model was not trained on fully reversed chains, it performs at par with the significantly larger GPT-4 model (Fig. 5). For other tasks, it often outperforms or matches the accuracy of billion-scale models like Gemini Pro and Phi-3. These results indicate that a model trained axiomatically can learn to reason about more complex causal structures from demonstrations of simple causal sequences. This suggests the potential of axiomatic training for reasoning over causal graphs.

H.2 Additional Results: Role of Data Diversity and Positional Encoding

Role of Positional Encodings. Comparing the performance of models wrt. choice of positional encoding, we find that models with no positional encoding generalize well to both longer lengths (upto chain length of 15) and complex, unseen graph structures, even though they are only trained on chains over 3-6 nodes. Models with `SPE` and `LPE` also perform well on longer chains but poorly when node names' length increases, even for chains with smaller number of nodes (Figure 2). This generalization failure using SPE and LPE highlights the models' inability to handle minor perturbations in the sequences present in their training set. Moreover, SPE does not perform well across different structural dimensions like branching, and order-based settings such as shuffling and reversal. Learnable PE performs well up to 9-length linear chains but drops sharply afterward. Overall, our results extend earlier work on the utility of `NoPE` [13, 10] to the task of understanding causal sequences and generalizing to both longer length and complex structure at test time.

Note that for the evaluation setup with randomly flipped edges, all types of PEs perform well. This may be due to the fact that the maximum effective length of a directed path between any two nodes is very small in randomly flipped sequences (much smaller than that of sequential causal chains) because the probability of an edge being forward directed is 0.5, and as we move across all sequential edges, the probability keeps on decreasing as 0.5^l where l is length of path.

Importance of Data Perturbations: In addition to the positional encoding, diversity of the sequences in train data plays an important role. Model trained on only causal chains (OCC) generalize to longer chains (Table 5) but not to other DAG structures (see Figure 3 for edge flip, Figure **??** for reversal, Table 4 for branching). Models trained on TS1 or TS2 generalize across all scenarios, including random flip, order permutations, and branching; thus highlighting the impact of incorporating variability at the edge level through random flipping. However, across tasks, we find that `TS2` yields higher accuracy than `TS1`, even as `TS1` has more variations due to random flipping. This suggests that while perturbations aid structural generalization, excessive perturbations can hinder it (in particular, random flipping may decrease the length of available causal paths during training).