Computing Actual Causes for Neural Network Predictions under Structured Causal Inputs

Jannick Strobel, Muqsit Azeem, and Stefan Leue

University of Konstanz, Germany, firstname.lastname@uni-konstanz.de

Abstract. We study the problem of computing actual causes for neural network (NN) predictions under structured input dependencies. Existing explanation methods typically assume feature independence, which can produce misleading explanations when inputs are causally related. To address this limitation, we formalize explanations using Halpern and Pearl actual causality within Structural Causal Models (SCMs). We reduce the computation of actual causes to a NN verification problem by combining differentiable relaxations with branch-and-bound verification techniques. Our preliminary experiments indicate that the proposed method is effective and scalable for computing causal explanations of neural network predictions.

Keywords: Causality · Neural Networks · Formal Verification

1 Introduction

Computing Actual Causality. We address the problem of computing *actual causality*, as defined by by Halpern and Pearl [HP05, Hal16] and henceforth referred to as HP actual causality, for the explanation of predictions made by Neural Networks (NNs). By providing causal explanations for these predictions, we contribute to the goal of explainable and trustworthy AI [ZTLT21]. Most existing explanation methods for NN predictions rely on attribution scores typically under an implicit assumption of feature independence [SREG+25]. In practice, however, inputs often exhibit structured dependencies, and ignoring these can lead to misleading or even spurious explanations [VT20, WWZ+23]. HP actual causality provides a principled framework for identifying causes of specific outcomes, capturing counterfactual dependence, contingency reasoning, and minimality. These properties make it particularly well-suited for explaining individual predictions. Computing HP actual causes has so far primarily been studied in discrete, non-neural settings [LL13, CDF+22, IRP19, IP20, ACM26].

Overview of the Approach. We address this limitation by adopting a formal notion of explanation based on HP's *actual causality* to the neural setting. For that we model dependencies between input features of a given NN using Structural Causal Models (SCMs) [Hal16]. Given a fixed input instance for the NN, which corresponds to a context in the SCM, and the resulting NN prediction, we compute HP actual causes for that decision. We reduce the search for these

causes to a NN verification problem which we solve using a differentiable relaxation to determine candidate causes and branch-and-bound NN verification techniques [WZX+21, XZW+21] to explore possible interventions.

Related Work. Closest to our work is the approach proposed in [ACM26]. It computes continuous relaxations of causal models enabling gradient-based search for HP actual causes and improves scalability compared to symbolic approaches [IRP19, IP20]. However, [ACM26] yields only approximate solutions whereas we provide formal guarantees along three key dimensions: (i) minimality of the computed causes, (ii) minimality and sufficiency of the set of used contingencies, and (iii) the computation of all valid HP actual causes.

2 Background and Setup

Structural Causal Models ([Hal16]). An SCM is a tuple $\mathcal{M} = (U, V, F)$, where U are exogenous variables, V endogenous variables, and F a set of structural equations. A *context* u assigns values to all exogenous variables and determines a unique assignment to all endogenous variables. In this work, we consider binary SCMs, where $X_j \in \{0, 1\}$ and structural equations are defined using Boolean functions.

Actual Causality (Modified Definition according to [Hal16]). Given a model $\mathcal{M}$ and context u, a set of variables $X = x$ is an actual cause of an outcome φ if: (AC1) $X = x$ and φ hold in the actual context, (AC2) changing X under some contingency leads to $\neg\varphi$, and (AC3) X is minimal. We compute actual causes of NN predictions under structured input dependencies. We model dependencies using a Structural Causal Model (SCM) $M = (U, V, F)$. Instead of checking individual counterfactuals, we reason about *sets of possible inputs* induced by interventions and analyze them using NN verification.

Setup. A context u induces a factual assignment x to all variables, which serves as input to a neural network f with output $y = f(x)$. Our goal is to find minimal subsets of variables that are actual causes of this prediction. We consider a prediction change with respect to a decision threshold τ. We rewrite the boolean structure of the SCM using continuous arithmetic expressions which allows for reasoning on the SCM using continuous relaxations. An intervention on a candidate cause X (with contingencies restricted to their factual values) induces not a single counterfactual, but a *set* of possible inputs. We represent this set symbolically as a region.

3 Causality Computation Leveraging Verification

We present our algorithm computing HP actual causality according to conditions AC1–AC3 ([Hal16]), shown in Algorithm 1, which leverages neural network verification.

For a candidate cause, the induced region defines a set of admissible inputs, and causality reduces to verifying properties of the network over this region.

Algorithm 1: Find All Minimal Causes

Input: Intervenable nodes V, Maximum cause size $k_{\max}$, Threshold τ, NN f
Output: Set of minimal causes $\mathcal{C}^*$

$\mathcal{C}^* \leftarrow \emptyset$
for *each k-subset* $C \subseteq V$*,* $k = 1, \ldots, k_{\max}$ **do**
  if $\exists\, C' \in \mathcal{C}^*$ *s.t.* $C' \subsetneq C$ **then skip** `// minimality`
  $\mathcal{B}_0 \leftarrow \text{bound}_{\text{IA}}(\text{SCM}(C))$ `// feature bounds over contingency space`
  for *method* $\in \{IBP,\ CROWN\}$ **do**
    $[\ell, h],\ \alpha \leftarrow \text{method}(f, \mathcal{B}_0)$
    if $\ell \geq \tau$ **then skip**
    if $h < \tau$ **then** $\mathcal{C}^* \leftarrow \mathcal{C}^* \cup \{C\}$; **continue**
  $Q \leftarrow \{\mathcal{B}_0\}$ `// branch-and-bound queue`
  while $Q \neq \emptyset$ **do**
    $\mathcal{B} \leftarrow \text{pop}(Q)$; $[\ell, h] \leftarrow \text{IBP}(f,\ \text{bound}_{\text{IA}}(\text{SCM}, \mathcal{B}))$
    if $\ell \geq \tau$ **then prune**
    else if $h < \tau$ **then** $\mathcal{C}^* \leftarrow \mathcal{C}^* \cup \{C\}$ **break**
    else $Q \leftarrow Q \cup \text{split}(\mathcal{B},\ \arg\max_i\ \alpha_i)$
return $\mathcal{C}^*$

In particular, we check whether intervening on the candidate induces a change in the prediction over the induced region under admissible contingencies. We encode this as a property over the network outputs and analyze it using sound bounds. Given output bounds over the region, we can (i) rule out candidates if no input can change the prediction, (ii) accept candidates if all inputs flip the prediction, or (iii) refine the region to establish existence of a counterfactual. AC3 (minimality) is enforced by the search procedure.

We enumerate candidate causes up to size k in increasing size. For each candidate, we construct the induced region and analyze it using bound propagation as follows: 1) We first compute coarse bounds using Interval Bound Propagation (IBP) [GDS+19], which propagates input intervals through the network to obtain sound output bounds. 2) If the bounds show that the prediction cannot change, we discard the candidate; if they guarantee a change, we accept it. If inconclusive, we refine bounds using CROWN [ZWC+18], which computes tighter linear relaxations of the NN and provides sensitivity information (α) used to guide branching. 3) For the remaining cases, we apply branch-and-bound. We iteratively split the region by fixing additional variables, recompute bounds, and prune or refine accordingly.

Soundness. Algorithm 1 enumerates all minimal actual causes up to size k, as the size-ordered search explores all candidates while pruning only infeasible regions and supersets of discovered causes.

4 Experimental Results

Experiments. We conducted preliminary experiments on the Steal Master Key (SMK) [IP20], a parametrized SCM representing a security protocol combined with a neural network that takes all endogenous and exogenous variables as input and predicts a risk score. We compared the performance of our approach against three baseline methods. (1) Brute Force (BF): Exhaustive combinatorial search over all possible variable sets and witnesses. (2) Integer Linear Programming (ILP): We encoded the cause search based on the ILP encoding of [IP20] and encoded the neural network using Big-M constraints. We employed an MINLP solver [HBB+25] and searched for all feasible causes. (3) Actual Causes Identification (ACI): A heuristic search-based approach [RDD25].

Results. The experimental results show that in the majority of cases, our method outperforms the baseline approaches with respect to the NN size, the maximum cause size and the number of users in the SMK benchmark. While BF and ILP-based methods quickly become intractable, and ACI often returns only partial causes, our approach consistently computes complete causes.

5 Conclusion

We compute actual causes of neural network predictions under structured causal inputs using branch-and-bound and neural network verification, with formal guarantees. Future work includes evaluation on larger networks and real-world datasets, and integrating gradient-based candidate generation with verification-based refinement.

References

[ACM26] Kaveh Aryan, Hana Chockler, and Mohammad Reza Mousavi. Differentiable causal search. In *Fifth Conference on Causal Learning and Reasoning*, 2026.

[CDF+22] Norine Coenen, Raimund Dachselt, Bernd Finkbeiner, Hadar Frenkel, Christopher Hahn, Tom Horak, Niklas Metzger, and Julian Siber. Explaining hyperproperty violations. In Sharon Shoham and Yakir Vizel, editors, *Computer Aided Verification*, pages 407–429, Cham, 2022. Springer International Publishing.

[GDS+19] Sven Gowal, Krishnamurthy Dvijotham, Robert Stanforth, Rudy Bunel, Chongli Qin, Jonathan Uesato, Relja Arandjelovic, Timothy Arthur Mann, and Pushmeet Kohli. Scalable verified training for provably robust image classification. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4841–4850, 2019.

[Hal16] Joseph Y. Halpern. *Actual Causality*. MIT Press, 2016.

[HBB+25] Christopher Hojny, Mathieu Besançon, Ksenia Bestuzheva, Sander Borst, Antonia Chmiela, João Dionísio, Leon Eifler, Mohammed Ghannam, Ambros Gleixner, Adrian Göß, Alexander Hoen, Rolf van der Hulst, Dominik

Kamp, Thorsten Koch, Kevin Kofler, Jurgen Lentz, Stephen J. Maher, Gioni Mexi, Erik Mühmer, Marc E. Pfetsch, Sebastian Pokutta, Felipe Serrano, Yuji Shinano, Mark Turner, Stefan Vigerske, Matthias Walter, Dieter Weninger, and Liding Xu. The SCIP Optimization Suite 10.0. Technical report, Optimization Online, November 2025.

[HP05] Joseph Y. Halpern and Judea Pearl. Causes and explanations: A structural-model approach. Part I: Causes. *Br. J. Philos. Sci.*, 56(4):843–887, 2005.

[IP20] Amjad Ibrahim and Alexander Pretschner. From Checking to Inference: Actual Causality Computations as Optimization Problems. volume 12302, pages 343–359. 2020.

[IRP19] Amjad Ibrahim, Simon Rehwald, and Alexander Pretschner. Efficiently Checking Actual Causality with SAT Solving, April 2019.

[LL13] Florian Leitner-Fischer and Stefan Leue. Causality checking for complex system models. In *VMCAI*, Lecture Notes in Computer Science, pages 248–267. Springer, 2013.

[RDD25] Samuel Reyd, Ada Diaconescu, and Jean-Louis Dessalles. Searching for actual causes: Approximate algorithms with adjustable precision. In *NeurIPS 2025 Workshop on CauScien: Uncovering Causality in Science*, 2025.

[SREG+25] Ahmed M. Salih, Zahra Raisi-Estabragh, Ilaria Boscolo Galazzo, Petia Radeva, Steffen E. Petersen, Karim Lekadir, and Gloria Menegaz. A perspective on explainable artificial intelligence methods: Shap and lime. *Advanced Intelligent Systems*, 7(1):2400304, 2025.

[VT20] Minh N. Vu and My T. Thai. Pgm-explainer: probabilistic graphical model explanations for graph neural networks. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA, 2020. Curran Associates Inc.

[WWZ+23] Xiang Wang, Yingxin Wu, An Zhang, Fuli Feng, Xiangnan He, and Tat-Seng Chua. Reinforced causal explainer for graph neural networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(2):2297–2309, 2023.

[WZX+21] Shiqi Wang, Huan Zhang, Kaidi Xu, Xue Lin, Suman Jana, Cho-Jui Hsieh, and J Zico Kolter. Beta-CROWN: Efficient bound propagation with per-neuron split constraints for complete and incomplete neural network verification. *Advances in Neural Information Processing Systems*, 34, 2021.

[XZW+21] Kaidi Xu, Huan Zhang, Shiqi Wang, Yihan Wang, Suman Jana, Xue Lin, and Cho-Jui Hsieh. Fast and Complete: Enabling complete neural network verification with rapid and massively parallel incomplete verifiers. In *International Conference on Learning Representations*, 2021.

[ZTLT21] Yu Zhang, Peter Tiňo, Ales Leonardis, and Ke Tang. A survey on neural network interpretability. *IEEE Trans. Emerg. Top. Comput. Intell.*, 5(5):726–742, 2021.

[ZWC+18] Huan Zhang, Tsui-Wei Weng, Pin-Yu Chen, Cho-Jui Hsieh, and Luca Daniel. Efficient neural network robustness certification with general activation functions. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, page 4944âĂŞ4953, Red Hook, NY, USA, 2018. Curran Associates Inc.