TOWARDS NEUROSYMBOLIC MECHANISTIC INTERPRETABILITY: THE CASE OF GRAPH NEURAL NETWORKS

Anonymous authors
Paper under double-blind review

ABSTRACT

Mechanistic interpretability (MI) aims to reverse-engineer subgraphs of neural networks into human-understandable circuits; yet, it lacks a formal language for specification and verification. This often results in leaky abstractions and unresolved theoretical problems with feature identifiability and universality. While symbolic AI generally avoids such pitfalls by leveraging the language of formal logic, it is naturally limited in both scope and scale. We propose a neurosymbolic synthesis, using MI to decompile latent representational superposition into discrete symbolic primitives, shifting interpretability from heuristic trial-and-error to a rigorous, logic-guided process. We demonstrate the feasibility of this synthesis through a case study on graph neural networks, utilizing their known logical expressiveness bounds to mathematically constrain the vocabulary of the respective latent feature space. Our experiments show that MI techniques, particularly sparse autoencoders, can reliably extract relational predicates, with targeted interventions verifying their logical compositions and interpretations. Ultimately, this work opens a path toward formal symbolic discovery in (relational) neural models.

1 INTRODUCTION

Safe deployment of deep learning models demands understanding of their decision-making processes, yet the internal causal logic of neural networks (NNs) remains mostly opaque (Confalonieri et al., 2021; Casper et al., 2024). The rapid scaling of these architectures, most prominently seen in large language models (LLMs), further obscures their internal processes and conceals the reasoning behind their predictions (Zhao et al., 2024). The field of mechanistic interpretability (MI) has emerged to address this by reverse-engineering neural subgraphs into human-understandable algorithms (Olah et al., 2020; Elhage et al., 2021; Conmy et al., 2023). Such transparency is particularly vital in high-stakes domains—for instance, verifying healthcare data and predictions can mitigate potentially life-impacting risks (Prenosil et al., 2025).

However, a significant limitation of current MI research is the lack of a formal language for interpretation specification (Sharkey et al., 2025). While MI offers tools to identify some internal functional motifs, it lacks a formal template for their semantics—a precise specification of what those motifs should represent. This often results in leaky abstractions[1] and unresolved theoretical hurdles, such as the problem of identifiability (Méloux et al., 2025), which make interpretations of model components difficult to verify or compare (Leemann et al., 2025; Yu & Ananiadou, 2024; Hadad et al., 2026).

In this exploratory work, we argue that these limitations can be mitigated by integrating MI with concepts from neurosymbolic AI (NeSy; Garcez & Lamb, 2023; Marra et al., 2024), which has a history of building on rigorous logical formalisms for prescribing internal properties of neural models. Particularly, we propose a unified perspective where NeSy serves as the logical view of neural computation, while MI provides the functional toolbox for probing latent representations of the models (Templeton et al., 2024; Lieberum et al., 2024; McDougall et al., 2025). This synthesis then facilitates a shift from the current empirical trial-and-error to a structured, logic-guided discovery process, as abstracted in Fig. 1.

[1] oversimplifications, hiding the underlying complexity of a neural mechanism (Spolsky, 2002)

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

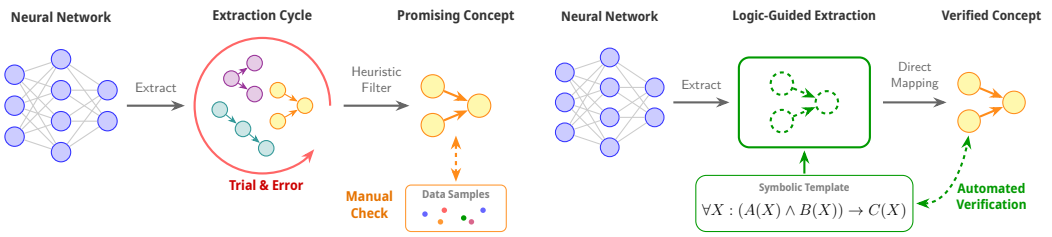


Figure 1: Standard MI pipeline (a) versus the proposed logic-guided pipeline (b). The latter replaces trial-and-error cycles with automated guidance by and verification against symbolic logic templates.

Our specific contributions center on the conceptual synthesis of MI and NeSy designed to decompile representational superposition and achieve verifiable symbol grounding. We instantiate this framework for graph neural networks (GNNs) by establishing a formal blueprint that utilizes their inherent $\mathcal{C}^2$ logical bounds (Grohe, 2021) to mathematically constrain post-hoc feature decomposition. Finally, we provide a feasibility demonstration showing the extraction and causal verification of relational predicates from a GNN using sparse autoencoders (SAEs; Huben et al., 2024).

2 Background

This section establishes the foundational principles of NeSy and MI, contrasting the prescriptive logic of the former with the emergent functional behaviors extracted by the latter as a theoretical basis for verifiable relational logic discovery in NNs.

2.1 Neurosymbolic AI: A Logical Perspective

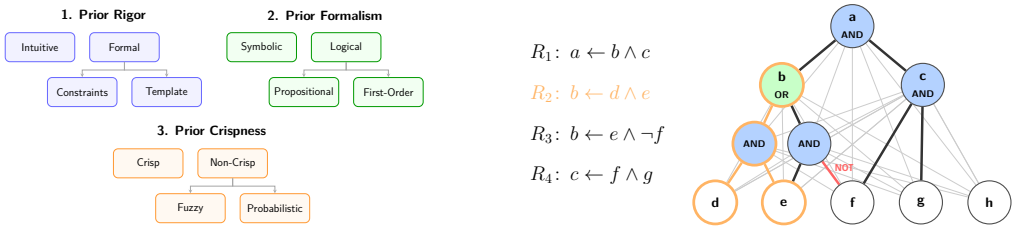
NeSy seeks to bridge deep learning's statistical power with the deductive rigor of symbolic logic (Garcez et al., 2019; Garcez & Lamb, 2023). Such hybrid systems are increasingly designed to augment frontier models with structured reasoning capabilities (Besold et al., 2021; Delvecchio et al., 2025; Obike et al., 2025). Rather than treating architectures as black-box function approximators, NeSy frameworks introduce symbolic priors that interact with neural components to prescribe system behavior (Odense & Garcez, 2025).

We characterize these priors by three dimensions (as seen in Fig. 2a): (i) *rigor*, discerning intuitive, often underdefined priors and formal priors, which can act as constraints or templates (Raedt et al., 2019); (ii) *formalism*, distinguishing between symbolic concepts in a broader sense (e.g., "dog" or "Golden Gate Bridge") and priors strictly governed by a formal language, such as propositional and first-order logic (FOL; Šír, 2020; Marra et al., 2024); and (iii) *crispness*, which dictates whether symbols are discrete or whether they should be interpreted as fuzzy or probabilistic truth values (Badreddine et al., 2022; Pryor et al., 2023). Naturally, out of the mentioned variants, it is the *formal logic* priors that allow researchers to properly analyze *interpretations*, such as verifying compositional correctness and reasoning about logical equivalence ($\phi \equiv \psi$ for formulae ϕ, ψ; Marra et al., 2024).

Beyond the nature of priors, NeSy systems can also be categorized by their functional interaction patterns (Kautz, 2022), ranging from `Symbolic[Neural]` architectures (e.g., AlphaGo Silver et al., 2016), where symbolic methods invoke neural components, through `Neural | Symbolic` systems (e.g., Neuro-Symbolic Concept Learner, Mao et al., 2019, or DeepProbLog, Manhaeve et al., 2018), which use NNs as perception tools for a symbolic engine, to $\texttt{Neural}_{\texttt{Symbolic}}$ systems (e.g., KBANN, Towell & Shavlik, 1993, or Logic Tensor Networks, Badreddine et al., 2022), where constraints are embedded directly into neural activations or loss functions.[2]

A foundational operational framework for conceptualizing these interactions is the *neurosymbolic learning cycle* (NSLC; Towell & Shavlik, 1993; Bader & Hitzler, 2005; Svatoš et al., 2019), formalizing the bidirectional translation between logic and neural weights through three iterative phases: *representation*, where symbolic knowledge initializes or constrains the neural architecture; *training*,

[2]For the discussion of all six classes, refer to the taxonomy of Kautz (2022).



a b

Figure 2: (a) Simplified taxonomy of symbolic prior properties in NeSy systems. (b) KBANN system with an exemplificative mapping between symbolic rules and neural network highlighted in orange.

where the model refines these rules against empirical data; and *extraction*, where the updated theory is read back from the trained weights (Bader & Hitzler, 2005).

Naturally, the pursuit of interpretability within NeSy models has historically focused on the extraction phase of NSLC, relying on *rule extraction* to derive symbolic clauses from trained weights (Andrews et al., 1995), as exemplified by the KBANN method (Fig. 2b) with its traceable mapping between symbolic rules and neural computation units (Towell & Shavlik, 1993). Modern approaches have expanded this to interpretable concept reasoning, enforcing intermediate layers grounded in human-understandable logic (Barbiero et al., 2023; Koh et al., 2020).

We argue that this mapping between neural weights and logical bounds provides the necessary formal blueprint that MI requires to move from purely post-hoc to prescriptive interpretations.

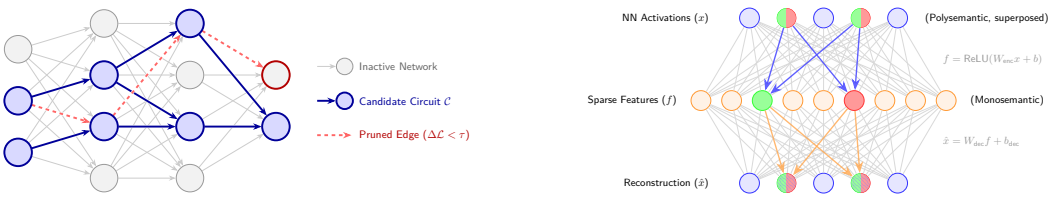
2.2 Mechanistic Interpretability: A Functional Perspective

MI generally aims to reverse-engineer neural networks into discrete, computational subgraphs of weights and activations—referred to as *circuits*—that implement specific algorithmic tasks (Olah et al., 2020; Elhage et al., 2021; Conmy et al., 2023). Primarily targeting transformer architectures (Vaswani et al., 2017), the models are treated as modular programs communicating via a shared residual stream, where each layer l contributes an additive update: $x_{final} = x_0 + \sum_{l=1}^{L} \Delta x_l$, with $\Delta x_l = \text{Layer}_l(x_{l-1})$, which allows researchers to independently interpret individual pathways/circuits (Elhage et al., 2021). To scale beyond manual analysis, automated circuit discovery utilizes edge pruning (Conmy et al., 2023; Fig. 3a) and its approximations (Syed et al., 2024), with the more recent attribution-graph techniques, referred to as circuit tracing (Hanna et al., 2025; Kamath et al., 2025), enabling understanding of complex feature interactions even in SotA models (Lindsey et al., 2025).

The core task in MI is understanding the latent representations of deep learning models. These representations naturally occur in *superposition*, which allows models to pack more features than available dimensions via non-orthogonal vectors (Elhage et al., 2022). To resolve the resulting *polysemanticity*, MI leverages dictionary learning (Fig. 3b)—mostly utilizing sparse autoencoders (SAEs; Shu et al., 2025)—to decompose dense activations into a higher-dimensional set of monosemantic "features" (Huben et al., 2024; Bricken et al., 2023). An SAE approximates an activation vector $x \in \mathbb{R}^d$ as a linear combination of monosemantic dictionary vectors $\mathbf{d}_i$: $x \approx \hat{x} = \sum_{i=1}^{M} f_i \mathbf{d}_i$, where $f_i \geq 0$ represents the sparse activation scalar of feature i. In alternative notation, the decoder computes $\hat{x} = W_{dec} f + b_{dec}$, where $W_{dec} \in \mathbb{R}^{d \times M}$ and $b_{dec} \in \mathbb{R}^d$ represent the decoder weights and biases respectively. To compute the sparse activations f_i, the SAE utilizes an encoder $f(x) = \text{ReLU}(W_{enc} x + b_{enc})$, where $W_{enc} \in \mathbb{R}^{M \times d}$ and $b_{enc} \in \mathbb{R}^M$. The weights W_{enc}, b_{enc}, W_{dec}, b_{dec} are updated via gradient descent on a λ-weighted sum of reconstruction ($\mathcal{L}_2$) and sparsity ($\mathcal{L}_1$) loss:

$$\mathcal{L} = \mathcal{L}_2 + \lambda \mathcal{L}_1 = \left\| x - \sum_{i=1}^{M} f_i \mathbf{d}_i \right\|_2^2 + \lambda \sum_{i=1}^{M} f_i. \tag{1}$$

This approach has been scaled to frontier models (Templeton et al., 2024; Lieberum et al., 2024; McDougall et al., 2025), and further performance gains have been achieved on some tasks utilizing alternatives to the sparsity penalty with top-k SAEs (Gao et al., 2025; Bussmann et al., 2024; Zhu



a b

Figure 3: Key methodologies in MI: (a) Automated circuit discovery through pruning and (b) Decomposition of activations into features using dictionary learning.

et al., 2026; described in App. B.4) and gated SAEs (Rajamanoharan et al., 2024), or by using specialized activation functions, such as JumpReLU (Rajamanoharan et al., 2025). Going a step further, *transcoders* have emerged as a specialized architecture that extends this decomposition by modeling the sparse transformations between multiple layers (Dunefsky et al., 2024; Paulo et al., 2025).

Beyond static decomposition, MI also investigates the "maturity" of representations across the model's depth—e.g., via the Logit Lens (nostalgebraist, 2020) and Tuned Lens (Belrose et al., 2023), inspecting how the model's prediction ability is refined layer by layer—while probing methods (Alain & Bengio, 2017; Belinkov, 2022) help to detect feature presence by training simple classifiers on latent representations. These perspectives are then complemented by the study of *universality* (Chughtai et al., 2023; Moisescu-Pareja et al., 2026), which asks whether different model instantiations converge to the same internal algorithms.

We argue that these tools provide the functional foundation for modern neurosymbolic synthesis, enabling feature extraction and verification against logical interpretations.

2.3 Graph Neural Networks and Logical Bounds

GNNs process non-Euclidean, graph-structured data by following a computational "message passing" scheme (Bronstein et al., 2021). Particularly, in message-passing GNNs, each node V updates its layer-l representation $\mathbf{h}_V^{(l)}$ by aggregating information from its neighbors $\mathcal{N}(V)$ (Kipf & Welling, 2017; Hamilton et al., 2017) as:

$$\mathbf{h}_V^{(l)} = \text{UPDATE}\left(\mathbf{h}_V^{(l-1)}, \text{AGG}\left(\left\{\phi\left(\mathbf{h}_U^{(l-1)}\right) \mid U \in \mathcal{N}(V)\right\}\right)\right), \tag{2}$$

where AGG is a permutation-invariant aggregation (e.g., sum) and ϕ is a learnable transformation.

Interestingly, the discriminative power of a standard message-passing GNN is rigorously upper-bounded by the 1-dimensional Weisfeiler-Leman (1-WL) graph isomorphism test (Xu et al., 2019). Further, the expressive power of the 1-WL test directly corresponds to the two-variable fragment of first-order logic with counting quantifiers, denoted as $\mathcal{C}^2$ (Barceló et al., 2020; Grohe, 2021). In this fragment, formulae are formed by using at most two distinct variables to evaluate structural relationships (e.g., $E(X, Y)$). Through this correspondence, the learnable neural projections of the GNN layers can be viewed as implementing these structural relationships, with the AGG operators instantiating the (local) counting quantifiers over them.

We argue that this expressiveness ceiling can be exploited as an example formal template for the proposed NeSy-MI synthesis, constraining the respective feature decomposition and circuit search to the above-mentioned class of logical formulae.

3 Synthesis of MI and NeSy

Synthesizing MI and NeSy requires alignment between the *functional view* of MI—how an NN performs computations through latent geometries—with the *logical view* of NeSy—what the NN computes through formal semantics. Note that this alignment fits well into the NSLC operations framework the NSLC operations framework (Bader & Hitzler, 2005; Sec. 2.1), where MI has the potential to operationalize multiple parts of the cycle (see Fig. 4), with the focus of this work being on

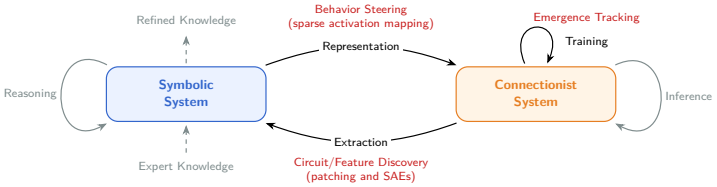


Figure 4: The NSLC with mechanistic instantiations of the relevant phases of the cycle in red.

the extraction and representation phases. In fact, circuit discovery and dictionary learning techniques from MI are exactly designed to recover "symbolic" concepts from the neural models, with the learned mappings possibly used to further influence the representations via steering; this, however, usually pertains only to the intuitive priors (Sec. 2.1), and the core challenge for our synthesis is to transcend the alignment to the formal priors. We propose to address this alignment through three primary mechanisms: (i) decompiling latent feature superposition to extract discrete symbolic primitives, (ii) lifting the extracted primitives into the relational domain governed by logical bounds, and (iii) causally verifying the alignment between latent neural representations and logical formulae.

3.1 Decompiling Superposition for Symbol Grounding

To achieve symbol grounding, we aim to use dictionary learning to decompile the representational *superposition* (Elhage et al., 2022; described in Sec. 2.2). We propose that SAEs (Huben et al., 2024) may serve well as the mathematical bridge to the logical domain by decomposing dense, superposed activations into a higher-dimensional set of monosemantic symbolic scalars f_i, with NeSy further mapping these functionally extracted scalars to *logical atoms*, predefined within some formal domain ontology (De Smet & De Raedt, 2025). Particularly, at the lowest level, SAE features would correspond to logical atoms, while at the intermediate level, functional circuits comprising specific neural weights would map these to *logical clauses*. Resolving superposition through SAEs thus serves as the prerequisite step for decompiling distributed neural activations into localist, discrete logic interpretations.

3.2 Causal Verification of Symbolic Grounding

The proposed synthesis aims to bridge the evaluative frameworks of both fields, where NeSy traditionally prioritizes *soundness* and formal validity (Renkhoff et al., 2024), while MI prioritizes empirical *faithfulness* (Heimersheim & Nanda, 2024). Let us use the GNN model class as an illustrative example. To establish that an SAE feature genuinely represents a specific $\mathcal{C}^2$ formula, mere correlation evaluation is insufficient. Instead, we may utilize a causal intervention toolkit to verify the symbolic grounding. By performing targeted MI interventions on the computational graph—such as feature ablation, counterfactual injection, and feature swapping—we can measure the resulting change in model predictions (e.g., the *flip rate*). If manipulating the specific SAE feature corresponding to a hypothesized logical rule causally dictates the network's output, we achieve a causally verified neurosymbolic extraction, providing strong empirical evidence—within the tested input distribution—that the continuous weights execute the interpreted discrete logic.

3.3 Relational Lifting and Logical Bounds

A notable limitation of current MI research is its central focus on attribute-based features in language models, which, from a logical perspective, correspond to a propositional language (Fig. 2a). To extend MI to relational[3] models, such as GNNs, we must perform a logical *lifting* (Šourek et al., 2018) of the representation. That is, rather than interpreting independent token properties, the extracted features must capture structural interdependencies between logical variables (e.g., corresponding to the nodes and their structured neighborhoods in graphs).

This is where the discussed theoretical properties of GNNs (Sec. 2.3), chosen as an example model class in this paper, benefit the proposed NeSy synthesis. Since the expressive power of standard message-passing GNNs is upper-bounded by the 1-WL test (Xu et al., 2019), the hypothesis space

[3]function-free subset of first-order logic (Fig. 2a)

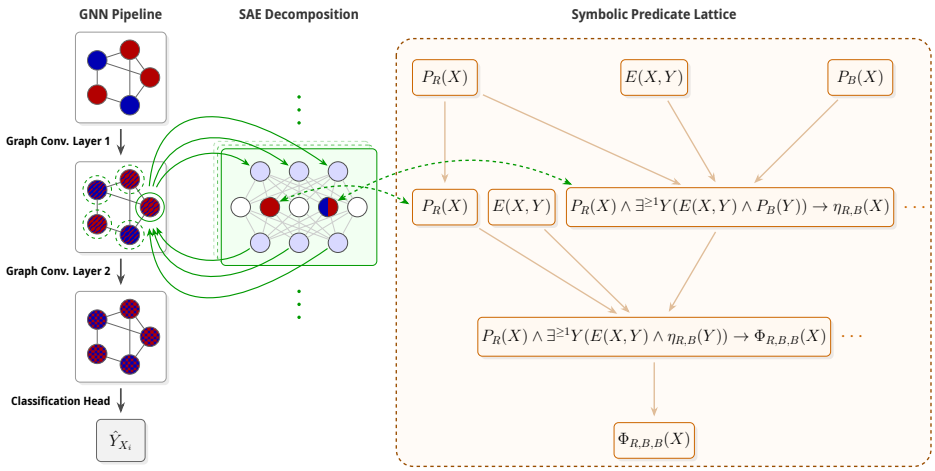


Figure 5: Neurosymbolic architecture for relational concept interpretability. A GNN pipeline (left) communicates with a symbolic predicate lattice (right) via the latent layer of an SAE (middle) encoding and decoding individual latent node representations.

for feature decomposition can be constrained to formulae from the $\mathcal{C}^2$ fragment of FOL (Sec. 2.3), directly forming the formal template for post-hoc interpretability. Such extracted SAE features then do not represent generic input (global graph) attributes, but can identify specific local relational properties of the nodes, such as their 1-hop or 2-hop neighborhood motifs (e.g., $\exists Y : Edge(X, Y) \wedge Color(Y, Blue)$). Thus, unlike the propositional features in language models, which typically represent isolated concepts (e.g., "Golden Gate Bridge"), these *relational* features directly ground the lifted relational logic formulae into the concrete connectivity of the input domain graphs (Grohe, 2021; Xu et al., 2019).

4 A Case Study: Relational Concept Extraction from GNNs

We demonstrate the proposed synthesis by intercepting the latent transformations of a GNN while mapping them to symbolic predicates in relational logic. Particularly, we align a standard functional computation—a multi-layer GNN—with a logical computation represented as a structured predicate lattice (as defined, e.g., by Muggleton & De Raedt, 1994). The neural and symbolic components then form a unified neurosymbolic interface, communicating via a mechanistic mapping facilitated by SAEs, as shown in Fig. 5.

Latent Decomposition via SAEs The process treats the GNN as a sequence of *relational* transformations, intercepting the continuous latent *node representations* generated by the convolutional layers across a corpus of structured graph inputs. Because these internal representations are naturally polysemantic, we train an SAE to disentangle the activation space. Applying the SAE decomposition to the representation of node V in the GNN's hidden layer l, we model the reconstruction as $\mathbf{h}_V^{(l)} \approx \sum_i f_{V,i}^{(l)} \mathbf{d}_i^{(l)}$, where $\mathbf{d}_i^{(l)}$ represents the decoder dictionary vectors mapping symbolic concepts back to the latent space, and $f_{V,i}^{(l)}$ denotes the sparse, monosemantic feature activations.

Predicate Lattice Alignment Once the latent space is decomposed, the resulting sparse feature activations $f_{V,i}^{(l)}$ are systematically aligned with the logical language. Specifically, we aim to bijectively tie these features to individual relational predicates $P_k(V)$ within the l-th layer of a predefined formal symbolic lattice. Particularly, for each predicate, we select the most discriminative sparse neuron i^* as its representative:

$$i^* = \arg\max_i \left(\frac{1}{|S^+|} \sum_{V \in S^+} f_{V,i}^{(l)} - \frac{1}{|S^-|} \sum_{V \in S^-} f_{V,i}^{(l)} \right), \tag{3}$$

where S^+ and S^- are the sets of nodes that satisfy and do not satisfy the target predicate, respectively. As the neural representations progress through the GNN layers and become capable of representing

more complex N-hop relational patterns, the corresponding lattice grows in complexity by logically composing predicates from previous layers.

Predicate Invention and $\mathcal{C}^2$ Pruning Notably, in the described form, the workflow also natively functions as a (scalable) relational *predicate invention* mechanism (Muggleton, 1994). While the known $\mathcal{C}^2$ logical limits of message-passing networks explicitly bound the set of computable predicates (Grohe, 2024), the potential lattice remains large. To manage this, we implement a dataset-driven pruning strategy (Svatoš et al., 2017) by identifying the neighborhood configurations realized in the data, and filtering out the unattainable predicates. This restricts our analysis only to the logically distinct relational patterns that can be observed in the experiment, such as $P_i(X) \wedge \exists^{=1} Y(E(X,Y) \wedge P_j(Y)) \wedge \exists^{=1} Z(E(X,Z) \wedge P_k(Z))$. By efficiently mapping the SAE's latent capacity to only the specific $\mathcal{C}^2$-definable predicates the model can empirically encounter, the pruning enables scalability of the framework without violating its role as a rigorous template.

Supplied vs. Discovered Information Note the division of labor between prior knowledge and discovery in the above workflow. Given a priori are the general structure of the hypothesis space—the (pruned) lattice induced by the model's logical bound—and the satisfaction sets S^+, S^-, computable for any candidate predicate directly from the input graphs. Discovered are the concrete predicates actually realized as monosemantic directions in the latent space, and their composition across layers. Notably, the SAE is trained without predicate supervision; the satisfaction sets serve only the post-hoc alignment, subsequently validated causally (Sec. 3.2). The pipeline thus performs logic-guided predicate alignment and verification, a stepping stone toward automating such symbolic discovery (Sec. 6.1).

4.1 Feasibility Demonstration

To validate the proposed synthesis, we run an end-to-end interpretability experiment on a GNN using an exhaustive synthetic dataset aimed at recovering a full set of logical compositions for a given relational target. This controlled setting is a deliberate methodological choice: verifying what extracted features represent requires known ground-truth logical structure and exhaustively testable interventions. In line with the MI practice of toy "model organisms" (Elhage et al., 2022; Chughtai et al., 2023; Nanda et al., 2023), our causal metrics constitute exhaustive checks over the entire input domain rather than sample estimates (App. A). Our goal is to use SAEs to (i) locate and decompose monosemantic features in latent GNN representations, (ii) map them to simple relational patterns ("building blocks" of the composed target), and (iii) verify this mapping via targeted interventions.

Relational Progression Suite We define a hierarchy of relational predicates where the depth corresponds to the number of message-passing steps required for their neural computation (Grohe, 2021). For colors $c_1, c_2, c_3 \in \{red, blue\}$ (abbreviated to R and B), we investigate a set of *depth-2 target predicates* $\{\Phi_{c_1,c_2,c_3}(X)\}$ over graph nodes. In our task, a node X is deemed positive (satisfies the target predicate) if it is of color c_1 and has a neighbor satisfying a *depth-1 "sub-predicate"* $\eta_{c_2,c_3}(Y)$. Formally:

$$\eta_{c_2,c_3}(Y) \equiv P_{c_2}(Y) \wedge \exists^{\geq 1} X(E(Y,X) \wedge P_{c_3}(X)) \tag{4}$$

$$\Phi_{c_1,c_2,c_3}(X) \equiv P_{c_1}(X) \wedge \exists^{\geq 1} Y(E(X,Y) \wedge \eta_{c_2,c_3}(Y)) \tag{5}$$

where P_c denotes the unary predicate for color c. We evaluate all four primary progressions: $\Phi_{R,R,B}$, $\Phi_{R,B,B}$, $\Phi_{R,B,R}$, and $\Phi_{R,R,R}$.[4] As seen in Eq. 4 and 5, the suite offers a natural decomposition of depth-2 predicates into depth-1 predicates, which is to be isolated via the SAE-based extraction. Their composition, implemented within the latent GNN layers, can then be verified against the (ground-truth) logical rules.

Experimental Setup We utilize the exhaustive dataset of all 354 connected non-isomorphic graphs with $n = 5$ nodes and two possible node colors ($c \in \{red, blue\}$). For each of the four progressions, we train a 2-layer GraphSAGE (Hamilton et al., 2017) binary node-level classifier with a hidden dimension of 128 on this dataset. The models consistently achieve perfect performance (1.0 test

[4]The four remaining three-node progressions are analogous up to a color swap.

Table 1: Relational Progression Suite: Primary SAE Causal Performance

Progression	Best λ	Necessity ($\Delta F1$)	Sufficiency (Single)	Specificity	CE Recovery
$\mathbf{\Phi_{R,R,B}}$	0.04	1.000	100.0%	91.5%	0.9675
$\mathbf{\Phi_{R,B,B}}$	0.05	0.9944	100.0%	100.0%	0.8100
$\mathbf{\Phi_{R,B,R}}$	0.01	0.9959	100.0%	100.0%	0.9933
$\mathbf{\Phi_{R,R,R}}$	0.01	1.000	95.7%	100.0%	1.0000

F1) within 35 epochs. For our analysis, we train high-dimensional SAEs ($d_{sae} = 2560$) on the first convolutional layer activations, where the depth-1 building blocks of the target composition (Eq. 4) must first be computed. We select primary SAE models for each progression by maximizing the balance of specificity and sufficiency (see App. B).

Causal Interventions and Metrics To quantify the functional role of the extracted features, we utilize three causal metrics: *causal necessity*, *causal sufficiency*, and *specificity*. We define an *intervention* as a targeted modification of neural activations to observe changes in the model output $M(V)$. Causal necessity measures the model's dependence on specific features via *zero ablation*, where target features are clamped to zero ($f_{target} = 0$), and is quantified by the resulting drop in F1-score. Causal sufficiency assesses if a feature can trigger a positive prediction on negative samples via *mean activation injection* of its mean activation μ into the latent stream. This is executed via *single-node injection* to verify if one "existential witness" is enough to flip the decision, and *multi-node injection* to test if multiple injected neighbors improve it further. Specificity measures how distinct a pattern is by *feature swapping* a task-relevant relational feature f_{target} with a task-irrelevant identity feature f_{id} on positive samples to verify the precision of the symbolic grounding. Finally, we measure *CE loss recovery*, quantifying how much of the original model's performance is restored by the SAE's reconstructed activations, relative to a baseline with the target layer's activations zero-ablated. Detailed formal definitions and experimental modes for each metric are provided in App. D.

Mapping and Functional Abstraction By aligning extracted features with the symbolic lattice, we observe that the GNN systematically partitions the continuous activation space into functional basis vectors corresponding to the relevant $\mathcal{C}^2$-definable predicates for every progression in the suite. Across all tasks, the SAE successfully recovered a consistent tripartite division of features into: (i) *Depth-1 Detectors* representing nodes satisfying the η sub-predicate (Eq. 4, e.g., $\eta_{R,B}(X)$ for $\Phi_{R,R,B}(X)$); (ii) *Negative Context* features representing the second nodes in the progression that fail the pattern test (e.g., $\eta_{B,R}(X)$ for $\Phi_{R,R,B}(X)$); and (iii) *Task Irrelevant* features mapping to nodes that cannot satisfy the task regardless of connectivity (e.g., $P_B(X)$ for any of the four progressions; for more examples of each category in full form, see Tab. 9 in App. B). We note that the disentanglement of Negative Context features from Depth-1 Detectors is particularly significant, as it indicates that the model represents the specific heterophily constraints required for sound logical deduction rather than simple color correlations (Méloux et al., 2025).

Causal Necessity and Conjunction We interpret *causal necessity* as an empirical test of the logical conjunction ($\wedge$) governing the target formula $\Phi(X)$ defined in Eq. 5. For $P_{c_1}(X) \wedge \exists^{\geq 1} Y(E(X,Y) \wedge \eta_{c_2,c_3}(Y))$, testing the conjunction means ablating either the $P_{c_1}(X)$ feature or the relevant subset of the features $\exists^{=N} Y(E(X,Y) \wedge \eta_{c_2,c_3}(Y))$ for $N \in \{1..5\}$. Across the entire progression suite, the zero-ablation of identified depth-1 features resulted in a near-total performance collapse ($\Delta F1 \approx 1.0$), as detailed in App. B. Within our exhaustive setting, this shows that the GNN's decision-making process is dependent on these specific relational units, empirically supporting the hypothesized mapping between the continuous neural weights and the formal logic clauses (Hadad et al., 2026).

Causal Sufficiency and Specificity Conversely, we utilize *causal sufficiency* and *specificity* to validate the existential quantifiers ($\exists^{\geq 1}$) and the monosemanticity of symbol grounding. For our primary SAE models, hallucinating a single existential witness via *single-node injection* was sufficient to trigger a positive classification in over 95% of cases, suggesting that the GNN functionally implements counting quantifiers over the extracted depth-1 basis (Grohe, 2021). Furthermore, our specificity tests yielded near-perfect flip rates, indicating that the GNN is sensitive to the precise relational structure of the predicate rather than generic node attributes.

Summary of Results The quantitative performance across the progression suite is summarized in Tab. 1. For all four evaluated patterns, the selected SAE models (best-performing λ) demonstrate near-total causal necessity ($\Delta F1$ up to 1.000). High causal sufficiency scores—exceeding 95.7% for single-node injections—confirm that these features effectively trigger the model's predictive logic. Furthermore, specificity scores between 91.5% and 100% support that the extracted features represent the precise relational predicates rather than generic node attributes, all while maintaining strong cross-entropy recovery ranging from 0.8100 to 1.0000. Stability across random seeds and SAE dictionary sizes is analyzed in App. B.3; grounding is also sensitive to SAE architecture—top-k variants reach superior reconstruction yet largely fail specificity (App. B.5)—which we treat as a finding in itself.

The experiments thus provide a closed-loop feasibility demonstration of the proposed MI-NeSy synthesis: continuous GNN activations can behave consistently with discrete relational logic programs, whose components can be identified and causally acted upon within our exhaustive setting.

5 Related Works

We contextualize our MI-NeSy synthesis within the emerging landscape of formal mechanistic discovery (for broader literature, refer to Sec. 2).

Recent MI research has begun to explore formal verification: Hadad et al. (2026) pioneers circuit discovery with soundness guarantees, complemented by formal minimal explanations (Boumazouza et al., 2026) and complexity-theoretic unifications of local and global explanations (Bassan et al., 2026). Further, efficient discovery of causal abstractions (Asiaee, 2026) and weight-based analysis increase interpretation robustness beyond activation-based patching (Golimblevskaia et al., 2025).

Weinberg (2025) views alignment of MI and NeSy as essential for reasoning; Helff et al. (2026) demonstrated reasoning in latent spaces via SAE-extracted features, Aswal & Hudelot (2025) utilized sparse concepts for safety guardrails, and Santis et al. (2026) distilled mechanistically extracted concepts into interpretable bottleneck models. SPARC (Nasiri-Sarvi et al., 2026) further enables cross-model interpretability, supporting Boolean logic learnability (Ranga et al., 2025) and symbolic regression (van Breda & Acar, 2026) research. While different in scope, these advancements support the outlined promise of logic-guided discovery in neural models.

6 Conclusion

We proposed a conceptual synthesis of mechanistic interpretability with neurosymbolic AI, facilitating a transition from post-hoc empirical observation to a logic-guided discovery process. This synthesis leveraged dictionary learning—specifically SAEs—to ground continuous latent representations into discrete symbolic primitives, instantiated via the inherent $\mathcal{C}^2$ logical bounds of message-passing GNNs to decompile internal representations into verifiable relational predicates (Grohe, 2021; Xu et al., 2019). Ultimately, this work opens the door for rigorous verification of internal model behaviors, determining whether (relational) neural models rely on intended logical patterns rather than unintended correlations (Prenosil et al., 2025; Hadad et al., 2026).

6.1 Limitations and Future Work

While our synthesis provides a formal path for interpretability, grounding latent representations in discrete symbolic primitives remains an open problem. Additionally, the current GNN study is limited by its low-capacity synthetic setting; while this "toy" problem served as a necessary unit test for verified symbol grounding (App. A), further research is required to evaluate scaling to larger models and more complex relational domains.

Scalability faces further hurdles: as GNN depth increases, the $\mathcal{C}^2$ lattice grows combinatorially (Grohe, 2024); our dataset-driven pruning mitigates this, but larger vocabularies will require lossy pruning (Svatoš et al., 2017) and frontier models more general methods like transcoders (Dunefsky et al., 2024). Finally, our verification depends on predefined logical templates; automating model-agnostic predicate invention (Muggleton, 1994) remains the primary long-term challenge.

AI USE STATEMENT

In this work, we used generative AI tools to aid and polish the writing (grammar checking, stylistic editing, and refinement), to draft parts of individual sections, and to assist with research execution—namely the implementation of standard methods and data processing/filtering code. We have not used generative AI tools for retrieval and discovery of related work, for generating synthetic datasets (our dataset is produced by programmatic enumeration of all connected five-node graphs), or for proving mathematical claims. We have reviewed all AI-assisted work: AI-assisted code was verified and tested for correctness by the authors, and all drafted or edited text was checked for factual and technical accuracy. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work uses only small-scale synthetic graph data and does not involve human subjects, personal data, or scraped datasets. We discuss potential societal impacts, including the dual-use risk of precise reverse-engineering of causal logic, in App. E.

REPRODUCIBILITY STATEMENT

Reproducibility of our experiments is supported by App. C, which details the dataset construction, training configurations, hyperparameters (App. C.2), and computational resources, as well as by the code provided in the supplementary materials, which includes the generation logic for the synthetic dataset and instructions for running the full pipeline. Formal definitions of all causal metrics are given in App. D.

REFERENCES

Guillaume Alain and Yoshua Bengio. Understanding intermediate layers using linear classifier probes. In *The Fifth International Conference on Learning Representations*, 2017.

Robert Andrews, Joachim Diederich, and Alan B Tickle. Survey and critique of techniques for extracting rules from trained artificial neural networks. *Knowledge-based systems*, 8(6):373–389, 1995.

Amir Asiaee. Efficient discovery of approximate causal abstractions via neural mechanism sparsification. *arXiv preprint* `arXiv:2602.24266`, 2026.

Darpan Aswal and Céline Hudelot. Conceptguard: Neuro-symbolic safety guardrails via sparse interpretable jailbreak concepts. *arXiv preprint* `arXiv:2508.16325`, 2025.

Sebastian Bader and Pascal Hitzler. Dimensions of neural-symbolic integration-a structured survey. *arXiv preprint cs/0511042*, 2005.

Samy Badreddine, Artur d'Avila Garcez, Luciano Serafini, and Michael Spranger. Logic tensor networks. *Artificial Intelligence*, 303:103649, 2022.

Pietro Barbiero, Gabriele Ciravegna, Francesco Giannini, Mateo Espinosa Zarlenga, Lucie Charlotte Magister, Alberto Tonda, Pietro Lio, Frederic Precioso, Mateja Jamnik, and Giuseppe Marra. Interpretable neural-symbolic concept reasoning. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 1801–1825. PMLR, 2023. URL `https://proceedings.mlr.press/v202/barbiero23a.html`.

Pablo Barceló, Egor V Kostylev, Mikael Monet, Jorge Pérez, Juan Reutter, and Juan-Pablo Silva. The logical expressiveness of graph neural networks. In *8th International Conference on Learning Representations (ICLR 2020)*, 2020.

Shahaf Bassan, Xuanxiang Huang, and Guy Katz. Unifying formal explanations: A complexity-theoretic perspective. In *The Fourteenth International Conference on Learning Representations*, 2026. URL `https://openreview.net/forum?id=kbps463Pdf`.

Yonatan Belinkov. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics*, 48(1):207–219, 2022.

Nora Belrose, Zach Furman, Logan Smith, Danny Halawi, Igor Ostrovsky, Lev McKinney, Stella Biderman, and Jacob Steinhardt. Eliciting latent predictions from transformers with the tuned lens. *arXiv preprint arXiv:2303.08112*, 2023.

Tarek R. Besold, Sebastian Bader, Howard Bowman, Pedro Domingos, Pascal Hitzler, Kai-Uwe Kühnberger, Luis C. Lamb, Priscila Machado Vieira Lima, Leo de Penning, Gadi Pinkas, et al. Neural-symbolic learning and reasoning: A survey and interpretation. In *Neuro-Symbolic Artificial Intelligence: The State of the Art*, pp. 1–51. IOS Press, 2021.

Ryma Boumazouza, Raya Elsaleh, Melanie Ducoffe, Shahaf Bassan, and Guy Katz. FAME: Formal abstract minimal explanation for neural networks. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=VJkNqJJAhV.

Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, Robert Lasenby, Yifan Wu, Shauna Kravec, Nicholas Schiefer, Tim Maxwell, Nicholas Joseph, Zac Hatfield-Dodds, Alex Tamkin, Karina Nguyen, Brayden McLean, Josiah E Burke, Tristan Hume, Shan Carter, Tom Henighan, and Christopher Olah. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*, 2023. URL https://transformer-circuits.pub/2023/monosemantic-features/index.html.

Michael M Bronstein, Joan Bruna, Taco Cohen, and Petar Veličković. Geometric deep learning: Grids, groups, graphs, geodesics, and gauges. *arXiv preprint arXiv:2104.13478*, 2021.

Bart Bussmann, Patrick Leask, and Neel Nanda. Batchtopk sparse autoencoders. In *NeurIPS 2024 Workshop on Scientific Methods for Understanding Deep Learning*, 2024. URL https://openreview.net/forum?id=d4dpOCqybL.

Stephen Casper, Carson Ezell, Charlotte Siegmann, Noam Kolt, Taylor Lynn Curtis, Benjamin Bucknall, Andreas Haupt, Kevin Wei, Jérémy Scheurer, Marius Hobbhahn, et al. Black-box access is insufficient for rigorous ai audits. In *Proceedings of the 2024 ACM Conference on Fairness, Accountability, and Transparency*, pp. 2254–2272, 2024.

Bilal Chughtai, Lawrence Chan, and Neel Nanda. A toy model of universality: Reverse engineering how networks learn group operations. In *International Conference on Machine Learning*, pp. 6243–6267. PMLR, 2023.

Roberto Confalonieri, Ludovik Coba, Benedikt Wagner, and Tarek R Besold. A historical perspective of explainable artificial intelligence. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 11(1):e1391, 2021.

Arthur Conmy, Augustine Mavor-Parker, Aengus Lynch, Stefan Heimersheim, and Adrià Garriga-Alonso. Towards automated circuit discovery for mechanistic interpretability. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 16318–16352. Curran Associates, Inc., 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/34e1dbe95d34d7ebaf99b9bcaeb5b2be-Paper-Conference.pdf.

Lennert De Smet and Luc De Raedt. Defining neurosymbolic ai. *arXiv preprint arXiv:2507.11127*, 2025.

Giovanni Pio Delvecchio, Lorenzo Molfetta, and Gianluca Moro. Neuro-symbolic artificial intelligence: a task-directed survey in the black-box models era. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence*, IJCAI '25, 2025. ISBN 978-1-956792-06-5. doi: 10.24963/ijcai.2025/1157.

Jacob Dunefsky, Philippe Chlenski, and Neel Nanda. Transcoders find interpretable llm feature circuits. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 24375–24410. Curran Associates, Inc., 2024. doi: 10.52202/079017-0768. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/2b8f4db0464cc5b6e9d5e6bea4b9f308-Paper-Conference.pdf.

Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Nova DasSarma, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 2021. URL https://transformer-circuits.pub/2021/framework/index.html.

Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, Roger Grosse, Sam McCandlish, Jared Kaplan, Dario Amodei, Martin Wattenberg, and Christopher Olah. Toy models of superposition. *Transformer Circuits Thread*, 2022. URL https://transformer-circuits.pub/2022/toy_model/index.html.

Matthias Fey, Jinu Sunil, Akihiro Nitta, Rishi Puri, Manan Shah, Blaž Stojanovič, Ramona Bendias, Alexandria Barghi, Vid Kocijan, Zecheng Zhang, Xinwei He, Jan Eric Lenssen, and Jure Leskovec. Pyg 2.0: Scalable learning on real world graphs. In *Temporal Graph Learning Workshop @ KDD 2025*, 2025. URL https://openreview.net/forum?id=DHHLkQvWqs.

Leo Gao, Tom Dupre la Tour, Henk Tillman, Gabriel Goh, Rajan Troll, Alec Radford, Ilya Sutskever, Jan Leike, and Jeffrey Wu. Scaling and evaluating sparse autoencoders. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=tcsZt9ZNKD.

Artur d'Avila Garcez, Marco Gori, Luis C Lamb, Luciano Serafini, Michael Spranger, and Son N Tran. Neural-symbolic computing: An effective methodology for principled integration of machine learning and reasoning. *arXiv preprint arXiv:1905.06088*, 2019.

Artur d'Avila Garcez and Luis C Lamb. Neurosymbolic ai: The 3 rd wave. *Artificial Intelligence Review*, 56(11):12387–12406, 2023.

Elena Golimblevskaia, Aakriti Jain, Bruno Puri, Ammar Ibrahim, Wojciech Samek, and Sebastian Lapuschkin. Circuit insights: Towards interpretability beyond activations. *arXiv preprint arXiv:2510.14936*, 2025.

Martin Grohe. The logic of graph neural networks. In *36th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2021, Rome, Italy, June 29 - July 2, 2021*, pp. 1–17. IEEE, 2021. doi: 10.1109/LICS52264.2021.9470677. URL https://doi.org/10.1109/LICS52264.2021.9470677.

Martin Grohe. The descriptive complexity of graph neural networks. *TheoretiCS*, 3, 2024.

Itamar Hadad, Guy Katz, and Shahaf Bassan. Formal mechanistic interpretability: Automated circuit discovery with provable guarantees. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=Timsb74vIY.

Aric A Hagberg, Daniel A Schult, and Pieter J Swart. Exploring network structure, dynamics, and function using networkx. In *Proceedings of the Python in Science Conference*, pp. 11–15. SciPy, 2008.

Will Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. *Advances in neural information processing systems*, 30, 2017.

Michael Hanna, Mateusz Piotrowski, Jack Lindsey, and Emmanuel Ameisen. Circuit-tracer: A new library for finding feature circuits. In *Proceedings of the 8th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP*, pp. 239–249, 2025.

Stefan Heimersheim and Neel Nanda. How to use and interpret activation patching. *arXiv preprint arXiv:2404.15255*, 2024.

Lukas Helff, Ruben Härle, Wolfgang Stammer, Felix Friedrich, Manuel Brack, Antonia Wüst, Hikaru Shindo, Patrick Schramowski, and Kristian Kersting. Activationreasoning: Logical reasoning in latent activation spaces. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=gGJh5AZTG7.

Robert Huben, Hoagy Cunningham, Logan Riggs Smith, Aidan Ewart, and Lee Sharkey. Sparse autoencoders find highly interpretable features in language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=F76bwRSLeK.

Harish Kamath, Emmanuel Ameisen, Isaac Kauvar, Rodrigo Luger, Wes Gurnee, Adam Pearce, Sam Zimmerman, Joshua Batson, Thomas Conerly, Chris Olah, and Jack Lindsey. Tracing attention computation through feature interactions. *Transformer Circuits Thread*, 2025. URL https://transformer-circuits.pub/2025/attention-qk/index.html.

Henry Kautz. The third ai summer: Aaai robert s. engelmore memorial lecture. *Ai magazine*, 43(1): 105–125, 2022.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations (ICLR 2015)*, 2015.

Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=SJU4ayYgl.

Pang Wei Koh, Thao Nguyen, Yew Siang Tang, Stephen Mussmann, Emma Pierson, Been Kim, and Percy Liang. Concept bottleneck models. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 5338–5348. PMLR, 2020. URL https://proceedings.mlr.press/v119/koh20a.html.

Chaitanya Kumar Kolli. Hybrid neuro-symbolic models for ethical ai in risk-sensitive domains. *arXiv preprint arXiv:2511.17644*, 2025.

Tobias Leemann, Alina Fastowski, Felix Pfeiffer, and Gjergji Kasneci. Attention mechanisms don't learn additive models: Rethinking feature importance for transformers. *Transactions on Machine Learning Research*, 2025. ISSN 2835-8856. URL https://openreview.net/forum?id=yawWz4qWkF.

Tom Lieberum, Senthooran Rajamanoharan, Arthur Conmy, Lewis Smith, Nicolas Sonnerat, Vikrant Varma, Janos Kramar, Anca Dragan, Rohin Shah, and Neel Nanda. Gemma scope: Open sparse autoencoders everywhere all at once on gemma 2. In *Proceedings of the 7th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP*, pp. 278–300, Miami, Florida, US, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.blackboxnlp-1.19. URL https://aclanthology.org/2024.blackboxnlp-1.19/.

Jack Lindsey, Wes Gurnee, Emmanuel Ameisen, Brian Chen, Adam Pearce, Nicholas L. Turner, Craig Citro, David Abrahams, Shan Carter, Basil Hosmer, Jonathan Marcus, Michael Sklar, Adly Templeton, Trenton Bricken, Callum McDougall, Hoagy Cunningham, Thomas Henighan, Adam Jermyn, Andy Jones, Andrew Persic, Zhenyi Qi, T. Ben Thompson, Sam Zimmerman, Kelley Rivoire, Thomas Conerly, Chris Olah, and Joshua Batson. On the biology of a large language model. *Transformer Circuits Thread*, 2025. URL https://transformer-circuits.pub/2025/attribution-graphs/biology.html.

Robin Manhaeve, Sebastijan Dumancic, Angelika Kimmig, Thomas Demeester, and Luc De Raedt. Deepproblog: Neural probabilistic logic programming. In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper_files/paper/2018/file/dc5d637ed5e62c36ecb73b654b05ba2a-Paper.pdf.

Jiayuan Mao, Chuang Gan, Pushmeet Kohli, Joshua B. Tenenbaum, and Jiajun Wu. The neuro-symbolic concept learner: Interpreting scenes, words, and sentences from natural supervision. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=rJgMlhRctm.

Giuseppe Marra, Sebastijan Dumančić, Robin Manhaeve, and Luc De Raedt. From statistical relational to neurosymbolic artificial intelligence: A survey. *Artificial Intelligence*, 328:104062, 2024.

Callum McDougall, Arthur Conmy, János Kramár, Tom Lieberum, Senthooran Rajamanoharan, and Neel Nanda. Gemma scope 2: Technical paper. Technical report, Google DeepMind, 9 2025.

Maxime Méloux, Silviu Maniu, François Portet, and Maxime Peyrard. Everything, everywhere, all at once: Is mechanistic interpretability identifiable? In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=5IWJBStfU7.

Gabriela Moisescu-Pareja, Gavin McCracken, Vincent Létourneau, Doina Precup, and Jonathan Love. Unifying mechanistic interpretations of neural networks trained on modular addition. In *Women in Machine Learning Workshop @ NeurIPS 2025*, 2026. URL https://openreview.net/forum?id=cCJoZxwh9s.

Stephen Muggleton. Predicate invention and utilization. *Journal of Experimental & Theoretical Artificial Intelligence*, 6(1):121–130, 1994.

Stephen Muggleton and Luc De Raedt. Inductive logic programming: Theory and methods. *The Journal of Logic Programming*, 19:629–679, 1994.

Neel Nanda, Lawrence Chan, Tom Lieberum, Jess Smith, and Jacob Steinhardt. Progress measures for grokking via mechanistic interpretability. In *The Eleventh International Conference on Learning Representations*, 2023.

Ali Nasiri-Sarvi, Hassan Rivaz, and Mahdi S. Hosseini. SPARC: Concept-aligned sparse autoencoders for cross-model and cross-modal interpretability. *Transactions on Machine Learning Research*, 2026. ISSN 2835-8856. URL https://openreview.net/forum?id=IJfvoc2GbZ.

nostalgebraist. Interpreting gpt: The logit lens. *lesswrong*, 2020. URL https://www.lesswrong.com/posts/AcKRB8wLmcNfGHRws/logit-lens.

Peter Godfrey Obike, Patience Usoro Usip, Edward Ndarake Udo, and Aniekan Joe Ananga. Neuro-symbolic reasoning: Performance, challenges, and benchmarks: A systematic literature review. *ABUAD Journal of Engineering and Applied Sciences*, 3(1):36–47, 2025.

Simon Odense and Artur d'Avila Garcez. A semantic framework for neurosymbolic computation. *Artificial Intelligence*, 340:104273, 2025.

Chris Olah, Nick Cammarata, Ludwig Schubert, Gabriel Goh, Michael Petrov, and Shan Carter. Zoom in: An introduction to circuits. *Distill*, 5, 3 2020. ISSN 2476-0757. doi: 10.23915/distill.00024.001. URL https://distill.pub/2020/circuits/zoom-in/?ref=cold-takes.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

Gonçalo Paulo, Stepan Shabalin, and Nora Belrose. Transcoders beat sparse autoencoders for interpretability. *arXiv preprint arXiv:2501.18823*, 2025.

George A Prenosil, Thilo K Weitzel, Sandra C Bello, Clemens Mingels, Giulia Manzini, Lorenz P Meier, Kuang-Yu Shi, Axel Rominger, and Ali Afshar-Oromieh. Neuro-symbolic ai for auditable cognitive information extraction from medical reports. *Communications Medicine*, 5(1):491, 2025.

Connor Pryor, Charles Dickens, Eriq Augustine, Alon Albalak, William Yang Wang, and Lise Getoor. Neupsl: neural probabilistic soft logic. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence*, pp. 4145–4153, 2023.

Luc De Raedt, Robin Manhaeve, Sebastijan Dumancic, Thomas Demeester, and Angelika Kimmig. Neuro-symbolic = neural + logical + probabilistic. In *NeSy@IJCAI*, 2019.

Senthooran Rajamanoharan, Arthur Conmy, Lewis Smith, Tom Lieberum, Vikrant Varma, Janos Kramar, Rohin Shah, and Neel Nanda. Improving dictionary learning with Gated Sparse Autoencoders. *arXiv preprint arXiv:2404.16014*, 2024.

Senthooran Rajamanoharan, Tom Lieberum, Nicolas Sonnerat, Arthur Conmy, Vikrant Varma, Janos Kramar, and Neel Nanda. Jumping ahead: Improving reconstruction fidelity with jumpreLU sparse autoencoders, 2025. URL https://openreview.net/forum?id=mMPaQzgzAN.

Sriram Ranga, Nandish Chattopadhyay, and Anupam Chattopadhyay. Learning nonlinearity of boolean functions: An experimentation with neural networks. *arXiv preprint arXiv:2502.01060*, 2025.

Justus Renkhoff, Ke Feng, Marc Meier-Doernberg, Alvaro Velasquez, and Houbing Herbert Song. A survey on verification and validation, testing and evaluations of neurosymbolic artificial intelligence. *IEEE Transactions on Artificial Intelligence*, 5(8):3765–3779, 2024.

Antonio De Santis, Schrasing Tong, Marco Brambilla, and Lalana Kagal. Learning concept bottleneck models from mechanistic explanations. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=gdEWoxhb70.

Lee Sharkey, Bilal Chughtai, Joshua Batson, Jack Lindsey, Jeffrey Wu, Lucius Bushnaq, Nicholas Goldowsky-Dill, Stefan Heimersheim, Alejandro Ortega, Joseph Isaac Bloom, Stella Biderman, Adrià Garriga-Alonso, Arthur Conmy, Neel Nanda, Jessica Mary Rumbelow, Martin Wattenberg, Nandi Schoots, Joseph Miller, William Saunders, Eric J Michaud, Stephen Casper, Max Tegmark, David Bau, Eric Todd, Atticus Geiger, Mor Geva, Jesse Hoogland, Daniel Murfet, and Thomas McGrath. Open problems in mechanistic interpretability. *Transactions on Machine Learning Research*, 2025. ISSN 2835-8856. URL https://openreview.net/forum?id=91H76m9Z94. Survey Certification.

Dong Shu, Xuansheng Wu, Haiyan Zhao, Daking Rai, Ziyu Yao, Ninghao Liu, and Mengnan Du. A survey on sparse autoencoders: Interpreting the internal mechanisms of large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 1690–1712, Suzhou, China, 2025. Association for Computational Linguistics. ISBN 979-8-89176-335-7. doi: 10.18653/v1/2025.findings-emnlp.89. URL https://aclanthology.org/2025.findings-emnlp.89/.

David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *nature*, 529(7587):484–489, 2016.

Gustav Šír. *Deep Learning with Relational Logic Representations / Hluboké Učení s Reprezentacemi v Relační Logice*. PhD thesis, Czech Technical University, 2020. AAI29117972.

Gustav Šourek, Vojtěch Aschenbrenner, Filip Železný, Steven Schockaert, and Ondřej Kuželka. Lifted Relational Neural Networks: Efficient Learning of Latent Relational Structures. *Journal of Artificial Intelligence Research*, 62:69–100, 2018. ISSN 1076-9757. URL http://jair.org/index.php/jair/article/view/11203.

Joel Spolsky. The law of leaky abstractions. In *Joel on Software: And on Diverse and Occasionally Related Matters That Will Prove of Interest to Software Developers, Designers, and Managers, and to Those Who, Whether by Good Fortune or Ill Luck, Work with Them in Some Capacity*, pp. 197–202. Springer, 2002.

Martin Svatoš, Gustav Šourek, Filip Železný, Steven Schockaert, and Ondřej Kuželka. Pruning hypothesis spaces using learned domain theories. In *International Conference on Inductive Logic Programming*, pp. 152–168. Springer, 2017.

Martin Svatoš, Gustav Šourek, and Filip Železný. Revisiting neural-symbolic learning cycle. In *14th International Workshop on Neural-Symbolic Learning and Reasoning*, 2019.

Aaquib Syed, Can Rager, and Arthur Conmy. Attribution patching outperforms automated circuit discovery. In *Proceedings of the 7th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP*, pp. 407–416, 2024.

Adly Templeton, Tom Conerly, Jonathan Marcus, Jack Lindsey, Trenton Bricken, Brian Chen, Adam Pearce, Craig Citro, Emmanuel Ameisen, Andy Jones, Hoagy Cunningham, Nicholas L Turner, Callum McDougall, Monte MacDiarmid, C. Daniel Freeman, Theodore R. Sumers, Edward Rees, Joshua Batson, Adam Jermyn, Shan Carter, Chris Olah, and Tom Henighan. Scaling monosemanticity: Extracting interpretable features from claude 3 sonnet. *Transformer Circuits Thread*, 2024. URL https://transformer-circuits.pub/2024/scaling-monosemanticity/index.html.

Geoffrey G Towell and Jude W Shavlik. Extracting refined rules from knowledge-based neural networks. *Machine learning*, 13(1):71–101, 1993.

Arco van Breda and Erman Acar. Explaining the explainer: Understanding the inner workings of transformer-based symbolic regression models. *arXiv preprint arXiv:2602.03506*, 2026.

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

Abraham Itzhak Weinberg. Neurosymbolic ai and mechanistic interpretability: Can they align in the artificial general intelligence era? *Available at SSRN 5151984*, 2025.

Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=ryGs6iA5Km.

Zeping Yu and Sophia Ananiadou. Neuron-level knowledge attribution in large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 3267–3280, Miami, Florida, USA, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.191. URL https://aclanthology.org/2024.emnlp-main.191/.

Haiyan Zhao, Hanjie Chen, Fan Yang, Ninghao Liu, Huiqi Deng, Hengyi Cai, Shuaiqiang Wang, Dawei Yin, and Mengnan Du. Explainability for large language models: A survey. *ACM Transactions on Intelligent Systems and Technology*, 15(2):1–38, 2024.

Xudong Zhu, Mohammad Mahdi Khalili, and Zhihui Zhu. Abstopk: Rethinking sparse autoencoders for bidirectional features. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=EEs6I4cO7S.

A On the Choice of an Exhaustive Synthetic Setting

Our feasibility demonstration deliberately targets a small, fully controlled domain. We expand here on why we consider this a methodological necessity for the present contribution, rather than a mere convenience.

Verification requires ground truth The central claim under test is that SAE features can be aligned with formal predicates and that this alignment is causally faithful. Assessing such a claim requires a domain where the complete logical structure of the task—the full lattice of realizable $\mathcal{C}^2$-definable predicates—is known by construction. In naturalistic domains, no such ground truth exists, and the evaluation of an interpretability method regresses into correlational plausibility checks—precisely the practice our synthesis seeks to move beyond. This mirrors the established MI methodology of "model organisms," where phenomena such as superposition (Elhage et al., 2022), universality (Chughtai et al., 2023), and emergent algorithmic structure (Nanda et al., 2023) were first isolated in small, controlled models before being sought at scale.

Table 2: $\Phi_{R,R,B}$ Experimental Results

λ	$\mathcal{L}_0$ Norm	Explained Var.	CE Loss Recovery	Necessity ($\Delta F1$)	Sufficiency	Specificity
0.01	2.61	0.9912	0.9998	1.0000	100.0%	0.0%
0.02	2.41	0.9747	0.9994	1.0000	100.0%	79.6%
0.03	2.13	0.9587	0.9773	1.0000	3.5%	86.6%
0.04	**1.40**	**0.9458**	**0.9675**	**1.0000**	**100.0%**	**91.5%**
0.05	1.34	0.9279	0.8457	1.0000	100.0%	0.0%

Table 3: $\Phi_{R,B,B}$ Experimental Results

λ	$\mathcal{L}_0$ Norm	Explained Var.	CE Loss Recovery	Necessity ($\Delta F1$)	Sufficiency	Specificity
0.01	2.42	0.9930	1.0000	0.2927	56.4%	52.2%
0.02	2.52	0.9828	0.9999	0.4473	93.2%	52.2%
0.03	1.82	0.9669	0.9115	1.0000	0.0%	58.2%
0.04	1.73	0.9586	0.9467	1.0000	6.0%	62.2%
0.05	**1.30**	**0.9410**	**0.8100**	**0.9944**	**100.0%**	**100.0%**

Exhaustiveness turns statistics into verification The dataset enumerates *all* 354 connected non-isomorphic five-node graphs with binary node colors; consequently, every realizable neighborhood configuration at this scale appears in the data, and no cherry-picking of favorable structures is possible. The causal metrics are therefore not sample estimates subject to sampling error or distribution shift, but exhaustive checks over the entire input domain: a feature that passes the necessity, sufficiency, and specificity tests here is verified against every input the model can encounter.

The setting can falsify The controlled setup does not guarantee positive results. The top-k SAE experiments (App. B.5) fail most specificity tests despite near-perfect reconstruction, and several sparsity configurations fail the sufficiency or specificity tests (App. B.2). These negative results demonstrate that the evaluation has discriminative power—success is a property of a particular, causally validated configuration, not an artifact of the setting. Moreover, any shortcut solution correlated with—but distinct from—the intended predicate would likewise be exposed by the intervention tests, the specificity test in particular.

A stepping stone toward scale Exhaustive testing is inherently bounded. For larger graphs and models, the same pipeline would need to move from exhaustive verification to sampling-based evaluation with probabilistic (e.g., PAC-style) guarantees, and from enumerated to lossily pruned predicate lattices (Sec. 6.1); higher-order GNNs may further introduce new identifiability hurdles in more complex logical fragments. Establishing the feasibility of verified symbol grounding in the exhaustive regime is, in our view, the prerequisite for trusting such extrapolations.

B SAE Robustness Study

B.1 Progression Suite Selection

Experiments across the relational progression suite show that the optimal sparsity coefficient λ is task-dependent. While reconstruction fidelity remains high across most configurations, our suite of symbolic interventions serves as the definitive objective function for identifying successful monosemantic grounding. In boundary-crossing tasks like $\Phi_{R,B,B}$, higher sparsity pressure is required to disentangle sub-predicates, whereas homophilous patterns like $\Phi_{R,R,R}$ achieve grounding at lower λ values.

B.2 SAE Results on Individual Patterns

The tables 2 to 5 summarize the trade-off between reconstruction quality and causal interpretability for each progression (for definitions of the result metrics, refer to App. D). Models that achieve optimal grounding (bolded λ) often differ from those with the highest explained variance. High

Table 4: $\Phi_{R,B,R}$ Experimental Results

λ	$\mathcal{L}_0$ Norm	Explained Var.	CE Loss Recovery	Necessity ($\Delta F1$)	Sufficiency	Specificity
0.01	**2.64**	**0.9922**	**0.9933**	**0.9959**	**100.0%**	**100.0%**
0.02	2.12	0.9778	0.8004	0.9325	0.0%	73.9%
0.03	2.18	0.9628	0.7632	0.2365	0.0%	72.9%
0.04	1.57	0.9550	0.9400	0.3329	0.0%	51.8%
0.05	1.65	0.9375	0.7847	0.3669	0.0%	0.0%

Table 5: $\Phi_{R,R,R}$ Experimental Results

λ	$\mathcal{L}_0$ Norm	Explained Var.	CE Loss Recovery	Necessity ($\Delta F1$)	Sufficiency	Specificity
0.01	**2.79**	**0.9944**	**1.0000**	**1.0000**	**95.7%**	**100.0%**
0.02	2.45	0.9723	0.9962	1.0000	18.4%	60.7%
0.03	2.03	0.9698	0.5639	1.0000	90.8%	99.3%
0.04	1.77	0.9503	0.0000	1.0000	63.1%	99.3%
0.05	1.50	0.9422	0.0000	1.0000	100.0%	100.0%

reconstruction scores at low λ often mask representational leakage, evidenced by low specificity scores in the $\Phi_{R,R,B}$ and $\Phi_{R,B,B}$ progression experiments.

B.3 Seed and Dictionary-Size Robustness

To assess the stability of the reported results, we ran the full pipeline (GNN and SAE training with subsequent interventions) across 10 random seeds. In this sweep, the GNN and SAE were seeded together, feature selection and all other hyperparameters were held fixed, and only the sparsity coefficient λ was optimized over. Tab. 6 reports the mean $\pm$ standard deviation of the causal metrics for each progression. Necessity and sufficiency remain high on average, while specificity is more variable for some progressions. Importantly, the variance is concentrated in a minority of poor runs rather than being uniformly distributed: for $\Phi_{R,B,B}$, 7 out of 10 runs achieve necessity ≥ 0.95 and specificity of 100%, and the two low-specificity runs (both at 52.2%) coincide with the low-necessity runs, suggesting shared poor SAE fits rather than a systematic method failure. A fuller protocol with an inner SAE-seed loop (selecting over independently trained SAEs per GNN) is expected to further reduce this spread, as a poor SAE initialization cannot be "optimized away" when seeded jointly with the GNN.

Table 6: Stability across random seeds (mean $\pm$ std. dev. over 10 runs)

Progression	Necessity ($\Delta F1$)	Sufficiency (Single)	Specificity	CE Recovery
$\Phi_{R,R,B}$	1.0000 ± 0.0000	$99.8\% \pm 0.7\%$	$93.7\% \pm 15.7\%$	0.9740 ± 0.0255
$\Phi_{R,B,B}$	0.7961 ± 0.2543	$97.3\% \pm 6.2\%$	$87.6\% \pm 19.6\%$	0.9109 ± 0.1227
$\Phi_{R,B,R}$	0.9488 ± 0.1099	$100.0\% \pm 0.0\%$	$88.1\% \pm 10.2\%$	0.9460 ± 0.0665
$\Phi_{R,R,R}$	0.9841 ± 0.0376	$96.1\% \pm 8.8\%$	$86.3\% \pm 14.0\%$	0.8674 ± 0.1992

Further, we analyzed the sensitivity to the SAE dictionary size on the $\Phi_{R,R,B}$ progression, sweeping the expansion factor of the SAE latent space relative to the GNN hidden dimension from $1\times$ to $30\times$ (with λ tuned per dictionary size; mean $\pm$ std. dev. across seeds). As shown in Tab. 7, the $20\times$ expansion—our primary configuration, chosen prior to this sweep—provides the best combination of high means and low variance across all four metrics. Notably, no expansion factor collapses on average: necessity stays at or above 0.89, and sufficiency and specificity remain above 83% on average across the full $1\times$–$30\times$ range. The main cost of moving away from $20\times$ is a higher seed-to-seed variance (particularly at $5\times/10\times$) rather than a drop in average performance, suggesting the method is not narrowly dependent on one specific dictionary size.

Table 7: Stability across SAE dictionary sizes ($\Phi_{R,R,B}$; mean $\pm$ std. dev. across seeds)

Expansion	Necessity ($\Delta F1$)	Sufficiency (Single)	Specificity	CE Recovery
$1\times$	0.9759 ± 0.0493	$98.1\% \pm 5.6\%$	$90.7\% \pm 11.8\%$	0.9574 ± 0.0854
$5\times$	0.8944 ± 0.2487	$92.4\% \pm 22.9\%$	$83.9\% \pm 29.4\%$	0.9794 ± 0.0390
$10\times$	1.0000 ± 0.0000	$91.4\% \pm 24.2\%$	$89.8\% \pm 23.9\%$	0.9846 ± 0.0230
$20\times$	1.0000 ± 0.0000	$99.8\% \pm 0.7\%$	$93.7\% \pm 15.7\%$	0.9740 ± 0.0255
$30\times$	0.9031 ± 0.2301	$96.6\% \pm 7.9\%$	$93.8\% \pm 9.4\%$	0.9740 ± 0.0714

Table 8: Top-k SAE Results across Bottleneck Sizes (for $\Phi_{R,R,B}$)

k Value	Exp. Var.	CE Loss Recovery	Necessity ($\Delta F1$)	Sufficiency	Specificity
1	0.8811	0.5992	1.0000	100.0%	0.0%
2	0.9951	0.9998	1.0000	100.0%	0.0%
3	0.9982	1.0000	1.0000	0.0%	0.0%
5	0.9993	1.0000	1.0000	0.0%	0.0%
10	1.0000	1.0000	0.9515	0.0%	49.3%
20	1.0000	1.0000	0.4233	0.0%	0.0%
50	1.0000	1.0000	0.1020	0.0%	0.0%

B.4 Top-k SAE Definition

In the top-k SAE variant, the $\mathcal{L}_1$ sparsity penalty is replaced by a hard bottleneck that retains only the k most active features. The encoder first computes the pre-activations $h = \text{ReLU}(W_{enc}x + b_{enc})$, from which the sparse feature vector f is derived: $f = \text{top-}k(h)$, where the top-k operator preserves the k largest components of the vector and sets the remaining $M - k$ elements to zero. Consequently, the reconstruction $\hat{x} = W_{dec}f + b_{dec}$ is optimized using only the reconstruction error, as the sparsity is structurally enforced: $\mathcal{L} = \mathcal{L}_2 = \|x - \hat{x}\|_2^2$.

B.5 Top-k SAE Results ($\Phi_{R,R,B}$ Progression)

To evaluate the impact of SAE architecture, we performed a representative set of experiments with top-k SAEs for the $\Phi_{R,R,B}$ progression. As shown in Tab. 8, while most top-k architectures achieve superior reconstruction fidelity (CE Loss Recovery), they fail most specificity tests compared to standard SAEs (see App. B.2). This suggests that continuous sparsity penalty should be preferred for relational discovery over hard top-k bottlenecks. We do not read these failures as invalidating the extraction approach: high reconstruction fidelity alone does not guarantee monosemantic relational grounding, and the choice of SAE architecture must be validated causally. Notably, this sensitivity appears more pronounced in our relational setting than in typical propositional MI tasks, suggesting that relational features place stronger demands on the SAE architecture.

B.6 Functional Grouping of Relational Features

Tab. 9 illustrates how the SAE abstracts realized structural patterns into functional categories. This provides evidence that the GNN's logic is organized around the symbolic primitives defined in Eq. 4.

Table 9: Functional Grouping of SAE Neurons ($\Phi_{R,R,B}$, $\lambda = 0.04$)

Functional Role	Symbolic Patterns Mapped (Counting Logic)
Depth-1 Detectors (η)	$P_R(x) \wedge \exists^{=1} y(E(x,y) \wedge P_B(y))$ $P_R(x) \wedge \exists^{\geq 2} y(E(x,y) \wedge P_B(y))$
Negative Context ($\neg\eta$)	$P_R(x) \wedge \exists^{\geq 2} y(E(x,y) \wedge P_R(y)) \wedge \neg\exists z(E(x,z) \wedge P_B(z))$ $P_R(x) \wedge \exists^{=1} y(E(x,y) \wedge P_R(y))$
Task Irrelevant	$P_B(x) \wedge$ Any Neighborhood

Table 10: Training Hyperparameters for GNN and SAE Models

Parameter	GNN (GraphSAGE)	SAE (Standard)
Hidden/Latent Dim	128	2560
Optimizer	Adam (Kingma & Ba, 2015)	Adam
Learning Rate	10^{-3}	10^{-2}
Batch Size	32	32
Sparsity Penalty (λ)	N/A	0.01 to 0.05
Training Epochs	max 200; with early stopping	3000
Loss Function	Binary Cross-Entropy	MSE + $\mathcal{L}_1$ Penalty

C Experimental Details and Reproducibility

The code for the feasibility demonstration (distributed under the CC BY 4.0 licence) can be found in the supplementary materials zip file. It includes a standalone Python script, README file and a file listing library dependencies (`pyproject.toml`). For instructions on running the script, refer to `README.md`. The details of the experimental settings are outlined below.

C.1 Dataset Configuration

The dataset comprises the set of 354 connected, non-isomorphic graphs with $n = 5$ nodes and binary node colors (red, $blue$). Target labels were assigned by evaluating the $\mathcal{C}^2$ formula for the relational tasks defined in Eq. 5. For the GNN training, the dataset was partitioned into training, validation, and test sets with a ratio of $70/15/15$. Graphs were verified for non-isomorphism to ensure no data leakage between splits. SAEs were trained on the entire dataset, as their purpose is to reconstruct the fixed activation space of the trained model, with feature validity subsequently verified through targeted causal interventions rather than held-out generalization metrics.

C.2 Hyperparameters, Setup, and Computational Resources

We summarize the training configurations for both the GNN and the Sparse Autoencoder in Tab. 10. The following software and hardware stacks were utilized:

- **Frameworks:** Experiments were implemented with in 2.11 (Paszke et al., 2019), Pytorch Geometric 2.7 (Fey et al., 2025), and NetworkX 3.4 (Hagberg et al., 2008).
- **Hardware:** Experiments were run on a CPU of a personal computer with 16GB RAM.
- **Execution Time:** Running the full pipeline (GNN/SAE training, interventions) for 5 different λ values required approx. 20 minutes for each of the relational patterns and 80 minutes in total.

D Formal Definition of Causal Metrics

Let M be the GNN, and f be the set of SAE features. We denote the model's prediction for a node V as $M(V) \in \{0, 1\}$.

D.1 CE Loss Recovery

CE loss recovery establishes a normalized scale to evaluate the fidelity of the SAE's reconstructions. We define $\mathcal{L}_{base}$ as the cross-entropy loss of the original, unmodified model, $\mathcal{L}_{zero}$ as the loss when activations at the target layer are completely zero-ablated, and $\mathcal{L}_{recon}$ as the loss when original activations are replaced with the SAE's reconstructed activations $\hat{x}$. The recovery metric is computed as:

$$\text{CE Recovery} = max(\frac{\mathcal{L}_{zero} - \mathcal{L}_{recon}}{\mathcal{L}_{zero} - \mathcal{L}_{base}}, 0) \tag{6}$$

A score of 1.0 represents perfect performance restoration, while 0.0 indicates performance equivalent to a complete information wipe.

D.2 Causal Necessity

Causal necessity measures the dependence of the model on a specific set of features f_{target}. It is defined as the change in F1-score following the ablation of those features:

$$\text{Necessity}(\Delta F1) = F1(M) - F1(M_{|f_{target}=0}) \tag{7}$$

where $M_{|f_{target}=0}$ represents the model's performance when the target SAE features are clamped to zero during the forward pass.

D.3 Causal Sufficiency

Causal sufficiency measures if the presence of a target feature f_{target} is enough to change the model's prediction on a set of negative samples S^-. To isolate the causal impact of the relational motif from the base node attributes, S^- is restricted to nodes that satisfy the root color constraint $P_{c_1}(X)$ but fail the overall predicate $\Phi(X)$. It is defined as the label flip rate:

$$\text{Sufficiency} = \frac{|\{V \in S^- \mid M_{|f_{target}=\mu}(V) = 1\}|}{|S^-|} \tag{8}$$

where $M_{|f_{target}=\mu}$ denotes the model output after injecting the mean activation μ of the target features into the hidden states of V's neighbors. In the context of relational models, we utilize two distinct injection modes to probe aggregation logic:

Single-Node Injection We inject the depth-1 feature into exactly *one* adjacent neighbor. This verifies if a single existential witness ($\exists Y : \eta(Y)$) satisfies the GNN's second-layer threshold for the target predicate $\Phi(X)$ (Grohe, 2024).

Multi-Node Injection We inject the feature into *all* adjacent neighbors simultaneously. This provides a maximally strong hallucinated signal. The disparity between single and multi-node success rates serves as a proxy for *conceptual resolution*; high-resolution features satisfy the threshold with a single witness, while distributed features require an aggregate signal to overcome internal bias.

D.4 Specificity

Specificity measures the conceptual resolution of extracted features by replacing a task-relevant relational feature f_{target} with a task-irrelevant identity feature f_{id} on a set of positive samples S^+. For our experiments, f_{target} corresponds to the Depth-1 Detector (e.g., $\eta_{R,B}(X)$) and f_{id} is chosen as the feature representing the base node color (e.g., $P_R(X)$) without relational context. It is defined as the flip rate to negative:

$$\text{Specificity} = \frac{|\{V \in S^+ \mid M_{|f_{target}=0, f_{id}=\mu_{id}}(V) = 0\}|}{|S^+|} \tag{9}$$

where μ_{id} is the mean activation of the identity feature. A high specificity score indicates that the model relies on the precise relational property rather than a generic attribute.

E Broader Impacts

Our framework facilitates a transition from heuristic trial-and-error to a rigorous, logic-guided process essential for safe AI deployment in high-stakes relational domains. By grounding neural features in formal logic, we aim for more auditable information extraction, which would help mitigate risks associated with model opacity in sensitive domains (Prenosil et al., 2025). As Hadad et al. (2026) note regarding formal mechanistic interpretability, such transparency is vital for provable guarantees.

Yet, we acknowledge that precise reverse-engineering of causal logic could be co-opted to design targeted adversarial attacks, which constitutes the most important potential negative impact of our work. Additional potential negative impact pertains to the computational resources needed for our interpretability pipeline—once scaled to frontier models, the cost of such experiments should be weighed against their benefits.

Despite this dual-use risk, we maintain that formal verification is a prerequisite for ensuring that high-stakes relational models operate within specified symbolic and ethical constraints (Kolli, 2025).