THE CAUSAL INFORMATION FLOW OF GRAPH NEURAL NETWORKS

Anonymous authors
Paper under double-blind review

ABSTRACT

Graph Neural Networks (GNNs) have achieved state-of-the-art performance in graph representation learning by exploiting statistical correlations. However, correlation-based learning fails to capture the underlying causal mechanisms, hampering GNN generalization under distribution shift in tasks like graph-level classification. Despite recent efforts towards developing Causal GNNs (CGNNs), state-of-the-art CGNN training involves complex multi-objective optimization, including objectives derived from the Mutual Information, a metric of correlational character. To address this limitation, we introduce a novel causal-by-design CGNN training paradigm based on the optimization of a single objective that is derived from the Causal Information Flow (CIF), an interventional metric. We evaluate our proposed CIF-based CGNN training paradigm against 2 foundational and 2 state-of-the-art CGNN training paradigms across 14 real and synthetic datasets coming from 4 different domains, using 3 classic Message Passing Neural Networks (MPNNs) and 3 state-of-the-art Graph Transformers (GTs). The results show that CIF-based training leads to competitive CGNN performance while optimizing a single objective.

1 INTRODUCTION

The advent of *Graph Neural Networks* (GNNs) (Kipf & Welling, 2016; Veličković et al., 2017; Hamilton et al., 2017; Xu et al., 2018; Rampášek et al., 2022; Pang et al., 2025; Zhuo et al., 2025) has driven crucial advancements in fields such as e-commerce (Xv et al., 2023; Niu et al., 2020; Feng et al., 2022; Wu et al., 2020), drug discovery (Cheung & Moura, 2020; Le, 2023; Abate et al., 2023; Wu et al., 2020), social-network analysis (Zhang et al., 2022; Xia et al., 2021; Tan et al., 2019), and financial crime detection (Motie & Raahemi, 2024; Li et al., 2023; Lin et al., 2024; Rico et al., 2026). *Message Passing Neural Networks* (MPNNs) (Kipf & Welling, 2016; Veličković et al., 2017; Hamilton et al., 2017; Xu et al., 2018), enable learning directly from graph-structured data rather than from Euclidean representations of the former. However, MPNNs face several limitations such as over-smoothing and over-squashing (Song et al., 2023). *Graph Transformers* (GTs) (Rampášek et al., 2022; Pang et al., 2025; Zhuo et al., 2025), help to alleviate said issues by extending the capabilities of traditional transformer architectures (Vaswani et al., 2017) to graph-structured data.

In spite of the effectiveness of GNNs in most graph representation learning tasks, recent studies have shown that in graph-level classification, the graph features that often determine a graph's label originate from causal substructures within the graph (Sui et al., 2022; Zhao et al., 2024; Chen et al., 2025). Nonetheless, modern GNNs (Kipf & Welling, 2016; Veličković et al., 2017; Hamilton et al., 2017; Xu et al., 2018; Rampášek et al., 2022; Pang et al., 2025; Zhuo et al., 2025) learn mainly by exploiting statistical correlations between the input graph features and the graph's label. Therefore, they generally learn to rely on spurious (i.e., non-causal) features in order to make label predictions. Consequently, they fail to capture the underlying causal mechanisms, limiting their generalization under distribution shift (Fan et al., 2022). *Causal GNNs* (CGNNs) (Sui et al., 2022; Wu et al., 2022; Fan et al., 2022; Chen et al., 2022; An et al., 2024; Zhao et al., 2024; Chen et al., 2025), emerged as an effective solution for learning generalizable GNNs that are robust under distribution shift.

Despite the promising results of CGNNs in *Out-Of-Distribution* (OOD) graph-level classification, the existing training paradigms for CGNNs (Sui et al., 2022; Wu et al., 2022; Fan et al., 2022; Chen et al., 2022; An et al., 2024; Zhao et al., 2024; Chen et al., 2025) involve complex multi-objective

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.

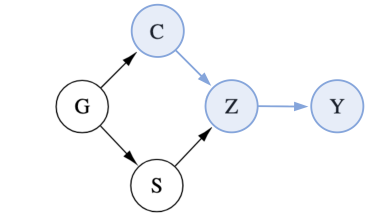

Figure 1: SCM for graph-level classification. Variable G denotes the input graph, C represents the causal features, S are the spurious (non-causal) features, Z is the latent representation of the graph, and Y is the label. The Causal Information Flow, denoted $I(C \rightarrow Y)$, is depicted in blue.

optimization, including objectives derived from the Mutual Information, a metric of correlational character. To address this limitation, we propose what is, to the best of our knowledge, the first *causal-by-design* CGNN training paradigm based on the optimization of a *single* optimization objective. Our optimization objective corresponds to a tractable lower bound of the *Causal Information Flow* (CIF), an interventional metric. Fixed a graph G, the Causal Information Flow, formally denoted as $I(C \rightarrow Y)$, allows measuring the extent to which the graph's label Y is determined based on its causal substructures C. Figure 1 provides a schematic depiction of CIF on the Structural Causal Model (SCM) for graph-level classification. Summarizing, our contributions are:

- The definition of the Causal Information Flow $I(C \rightarrow Y)$.
- The introduction of the first *single-objective causal-by-design* CGNN training paradigm.
- An extensive comparative evaluation against the state-of-the-art CGNN training paradigms.

2 Related Work

Causal Graph Neural Networks (CGNNs) are a model class designed to excel at OOD graph-level classification by relying on the extraction of causal sub-structures (Sui et al., 2022; Wu et al., 2022; Fan et al., 2022; Chen et al., 2022; An et al., 2024; Zhao et al., 2024; Chen et al., 2025). CGNNs follow the *Representation-Disentanglement-Prediction* (RDP) pipeline. As depicted in Figure 2, CGNNs first encode the input graph into the latent space (Representation). Subsequently, they separate or disentangle causal from spurious features in said representation space (Disentanglement). Finally, they make a label prediction based on the disentangled representation (Prediction). *Self-Explainable Graph Neural Networks* (XGNNs) (Dai & Wang, 2021; Miao et al., 2022; Yu et al., 2021; Liu et al., 2023; Chen et al., 2024; Deng & Shen, 2024) also follow the RDP pipeline. However, the latent space disentanglement yields explanatory and non-explanatory representations, rather than causal and spurious representations. XGNN learning is steered towards the generation of faithful explanations to accompany accurate predictions, rather than towards OOD generalization.

Despite the strong empirical performance of CGNNs in OOD graph-level classification, existing CGNN training paradigms combine a classification loss with one or more auxiliary terms. CAL (Sui et al., 2022) and ACE (Chen et al., 2025) include a back-door-adjusted interventional term, DIR (Wu et al., 2022) adds a penalty term across interventional distributions, and DisC (Fan et al., 2022) adds a bias-amplifying generalized cross-entropy term on a separate branch, while CIGA (Chen et al., 2022), CSIB (An et al., 2024), and ICL (Zhao et al., 2024) add terms derived from the Mutual Information. In all the aforementioned CGNN training paradigms, two or more optimization objectives must be balanced. To overcome this limitation, we introduce what is, to the best of our knowledge, the first *causal-by-design* CGNN training paradigm based on the optimization of a *single* optimization objective derived from the Causal Information Flow (CIF). The Causal Information Flow is an adaptation of the Information Flow (IF) (Ay & Polani, 2008) to the graph-level classification task. Information Flow (Ay & Polani, 2008) is the causal counterpart of Mutual Information. Information Flow-based losses have been proposed for post-hoc GNN explainer model selection (Lin et al., 2022). Nonetheless, to the best of our knowledge, the derivation of a CGNN training objective from the Information Flow (Ay & Polani, 2008) is unprecedented.

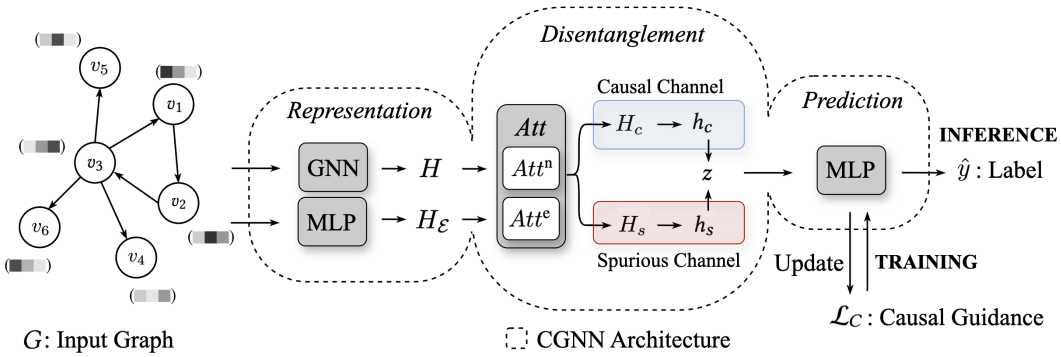

Figure 2: Representation-Disentanglement-Prediction (RDP) pipeline.

3 Preliminaries and Problem Statement

Let $G = (\mathcal{V}, \mathcal{E}, X, X_{\mathcal{E}})$ be an *attributed graph* where $\mathcal{V}$ is a set of $n \in \mathbb{N}$ nodes, and $\mathcal{E}$ is a set of $m \in \mathbb{N}$ edges. $X \in \mathbb{R}^{n \times d}$ denotes the node feature matrix, and $X_{\mathcal{E}} \in \mathbb{R}^{m \times d'}$ denotes the edge feature matrix, with $d, d' \in \mathbb{N}$. Further, $A \in \{0,1\}^{n \times n}$ refers to the adjacency matrix induced by $\mathcal{E}$, and $\mathcal{G}$ denotes the set of attributed graphs with label set $\mathcal{Y}$. Given a graph $G \in \mathcal{G}$ with label $y \in \mathcal{Y}$, an *invariant graph rationale* of G is a subgraph $G_c \subseteq G$ that has a causal relation with the label y.

Problem 1 (Out-of-Distribution Graph-Level Classification) *Given a collection of attributed graphs $\mathbb{G} \subseteq \mathcal{G}$ with corresponding labels $\mathbb{Y} \subseteq \mathcal{Y}$ as the training set, the task is to learn a CGNN $f : \mathcal{G} \to \mathcal{Y}$ capable of predicting the label of an Out-of-Distribution (OOD) test graph $G \in \mathcal{G} \setminus \mathbb{G}$. That is, a test graph in which the feature-label correlations present in the training set do not hold.*

4 Causal Graph Neural Networks (CGNNs)

CGNNs generalize under distribution shift by extracting causal sub-structures from the input graph. They follow the RDP pipeline schematized in Figure 2. We next describe each step in this pipeline.

4.1 Representation

Both nodes and edges of the input graph $G = (\mathcal{V}, \mathcal{E}, X, X_{\mathcal{E}})$ are assigned a latent representation:

$$H = \mathrm{GNN}(X, A, X_{\mathcal{E}}) \in \mathbb{R}^{n \times d''}, \qquad H_{\mathcal{E}} = \mathrm{MLP}(X_{\mathcal{E}}) \in \mathbb{R}^{m \times d''}, \tag{1}$$

where GNN is a graph encoder, MLP is a 2-layer perceptron with ReLU activation, and $d'' \in \mathbb{N}$ is the latent dimension. If G has no edge features, the representation of an edge $e = (u, v) \in \mathcal{E}$ is instead obtained by concatenating the rows u and v of H and projecting into $\mathbb{R}^{d''}$.

4.2 Disentanglement

Causal and spurious features are disentangled using node- and edge-level attention:

$$Att^{\mathrm{n}} = \mathrm{softmax}(\mathrm{GNNConv}(\mathrm{H}, \mathrm{A})) \in \mathbb{R}^{n \times 2}, \qquad Att^{\mathrm{e}} = \mathrm{softmax}(\mathrm{Lin}(H_{\mathcal{E}})) \in \mathbb{R}^{m \times 2}, \tag{2}$$

where GNNConv is a single GCN convolution (Kipf & Welling, 2016), and Lin is a linear layer. In both cases, the first and second columns yield the spurious and causal attention coefficients:

$$\alpha_s^{\mathrm{n}} = Att^{\mathrm{n}}_{:,0}, \quad \alpha_c^{\mathrm{n}} = Att^{\mathrm{n}}_{:,1}, \qquad \alpha_s^{\mathrm{e}} = Att^{\mathrm{e}}_{:,0}, \quad \alpha_c^{\mathrm{e}} = Att^{\mathrm{e}}_{:,1}. \tag{3}$$

The actual disentanglement step happens over the hidden node representations,

$$H'_s = \alpha_s^{\mathrm{n}} \mathbf{1}^\top \odot H \in \mathbb{R}^{n \times d''}, \quad H'_c = \alpha_c^{\mathrm{n}} \mathbf{1}^\top \odot H \in \mathbb{R}^{n \times d''}, \tag{4}$$

where $\mathbf{1} \in \mathbb{R}^{d''}$ is the vector of ones, $\top$ denotes transposition, and $\odot$ is the Hadamard product. The edge-level attention coefficients are incorporated through a single weighted GCN convolution,

$$H_s = \mathrm{GNNConv}(\mathrm{H}'_{\mathrm{s}}, \mathrm{A}, \alpha_{\mathrm{s}}^{\mathrm{e}}), \quad H_c = \mathrm{GNNConv}(\mathrm{H}'_{\mathrm{c}}, \mathrm{A}, \alpha_{\mathrm{c}}^{\mathrm{e}}), \tag{5}$$

and the disentangled representations are obtained via pooling:

$$h_s = \text{READOUT}\left(H_s\right), \quad h_c = \text{READOUT}\left(H_c\right). \tag{6}$$

For short, we denote by g^c_ω and g^s_ω the maps defined through Equations 1-6. That is,

$$h_c = g^c_\omega(G), \qquad h_s = g^s_\omega(G), \tag{7}$$

where G is the input graph, and $\omega \in \Omega$ denotes the parameter configuration.

4.3 Prediction

Finally, h_s and h_c are concatenated into $z = [h_c||h_s]$ and feed to the classification head:

$$\hat{y} = \text{softmax}(\text{MLP}(z)), \tag{8}$$

where MLP is a 2-layer perceptron with an ELU activation. For short, we refer to the prediction head defined in Equation (8) by ϕ_θ, where $\theta \in \Theta$ is the parameter configuration. We write $\phi_\theta(z)_y$ to denote the probability assigned by the prediction head to the label y.

5 A Causal-by-Design Training Paradigm for CGNNs

We propose a *causal-by-design* CGNN training paradigm based on the optimization of a *single* optimization objective derived from the Causal Information Flow (CIF), an interventional metric.

5.1 Theory of Causality

First, we provide a brief background on causality theory covering DAGs and back-door adjustment.

5.1.1 Directed acyclic graphs

Directed acyclic graphs (DAGs) serve as a powerful tool for modeling causal relations between variables (Pearl, 2009). Fixed a finite nonempty set of variables $\mathbf{V}$ and a set $\mathbf{E} \subseteq \mathbf{V} \times \mathbf{V}$ of directed connections between them, the graph $\mathbf{G} = (\mathbf{V}, \mathbf{E})$ serves as a model for the causal interactions between said variables. Namely, given $X, Y \in \mathbf{V}$, we include $(X, Y) \in \mathbf{E}$ if X causes Y, written $X \to Y$. Nodes X and Y are *adjacent*, symbolyzed as $X \sim Y$, if $X \to Y$ or $Y \to X$.

An ordered sequence of nodes $p = (X_1, \ldots, X_h)$ is called a *path* from X_1 to X_h, if $X_i \sim X_{i+1}$ for $i = 1, \ldots, h-1$. A path p in $\mathbf{G}$ is said to be *blocked* (Pearl, 2009) by a subset of variables $\mathbf{Z} \subseteq \mathbf{V}$ if one of the two following conditions hold: (1) The path p contains a *chain* $X \to Z \to Y$ or a *fork* $X \leftarrow Z \to Y$ such that the middle node $Z \in \mathbf{Z}$; or (2) The path p contains a *collider* $X \to Z \leftarrow Y$ such that neither Z or any of its descendants in $\mathbf{G}$ are in $\mathbf{Z}$.

5.1.2 The Back-Door Adjustment Formula

A set of variables $\mathbf{Z} \subseteq \mathbf{V}$ satisfies the *back-door criterion* (Pearl, 2009) relative to an ordered pair of variables (X, Y) in $\mathbf{G}$ if both the following conditions hold: (1) No node in $\mathbf{Z}$ is a descendant of X in $\mathbf{G}$; and (2) $\mathbf{Z}$ blocks every path between X and Y that contains an arrow into X in $\mathbf{G}$. As Theorem 5.1 shows, the back-door criterion is a powerful tool as it renders interventional distributions identifiable from observational ones.

Theorem 5.1 (The back-door adjustment formula (Pearl, 2009)) *If a set of variables $\mathbf{Z}$ satisfies the back-door criterion relative to (X, Y), then*

$$\mathbb{P}(Y|do(X = x)) = \sum_{\mathbf{z}} \mathbb{P}(Y|X = x, \mathbf{Z} = \mathbf{z})\mathbb{P}(\mathbf{Z} = \mathbf{z}), \tag{9}$$

where $do(\cdot)$ is the do operator.

5.2 The Causal Information Flow

Next, we define the *Causal Information Flow* (CIF). Fixed a graph $G \in \mathcal{G}$ with label $y \in \mathcal{Y}$ and a subgraph $G_c \subseteq G$, CIF allows quantifying the strength of the causal relation $G_c \to y$.

5.2.1 FROM STOCHASTIC INDEPENDENCE TO CAUSAL INDEPENDENCE

Two random variables X, Y are *stochastically independent* if,

$$\mathbb{P}(X,Y) = \mathbb{P}(X)\mathbb{P}(Y) \tag{10}$$
$$\iff \mathbb{P}(X|Y=y) = \mathbb{P}(X) \text{ for all } y \text{ with } \mathbb{P}(Y=y) > 0,$$
$$\iff \mathbb{P}(Y|X=x) = \mathbb{P}(Y) \text{ for all } x \text{ with } \mathbb{P}(X=x) > 0. \tag{11}$$

We focus on Equation (11) here: $\mathbb{P}(Y|X=x) = \mathbb{P}(Y)$ means that observing $X=x$ does not change the probability of observing Y. We define *causal independence* as the causal counterpart of Equation (11). Namely,

$$\mathbb{P}(Y|do(X=x)) = \mathbb{P}(Y) \text{ for all } x. \tag{12}$$

That is, setting X to x (i.e, $do(X=x)$), does not change the probability of observing Y. We note that, unlike stochastic independence, causal independence is directional. Equation (12) indicates that Y is causally independent of the interventions on X, and does not imply the converse. Our definition of causal independence is a special case of that proposed by Ay & Polani (2008). The interested reader can find the proof in Appendix A.1.

5.2.2 FROM STOCHASTIC DEPENDENCE TO CAUSAL DEPENDENCE

The *Mutual Information* between X and Y, denoted $I(X;Y)$, measures the stochastic dependence of X and Y as their deviation, in terms of the *Kullback–Leibler* (KL) *divergence*, from stochastic independence. That is, referring to Equation (10), $I(X;Y) := D_{KL}\left[\mathbb{P}(X,Y) \,||\, \mathbb{P}(X)\mathbb{P}(Y)\right]$. Equivalently, looking at Equation (11),

$$I(X;Y) = \mathbb{E}_{x\sim\mathbb{P}(X)}\left[D_{KL}\left[\mathbb{P}(Y|X=x) \,||\, \mathbb{P}(Y)\right]\right], \tag{13}$$

where $D_{KL}[\cdot||\cdot]$ denotes the KL divergence. We define *causal dependence* as the causal counterpart of Equation (13), replacing observational conditioning with interventional conditioning. Formally,

$$I(X \to Y) := \mathbb{E}_{x\sim\mathbb{P}(X)}\left[D_{KL}[\mathbb{P}(Y|do(X=x))||\mathbb{P}(Y)]\right]. \tag{14}$$

$I(X \to Y)$ is the *Information Flow* from X to Y, and it measures the strength of the causal relation $X \to Y$. In essence, it is the causal counterpart of the Mutual Information $I(X;Y)$, which measures the strength of the correlation between X and Y. Our definition of information flow differs from that of Ay & Polani (2008) in its reference distribution. Refer to Appendix A.2 for the details.

5.2.3 DEFINING THE CAUSAL INFORMATION FLOW

Figure 1 depicts the *Structural Causal Model* (SCM) describing the causal relationships between variables in graph representation learning. It consists of the input graph G, the graph's causal features C, the graph's non-causal (spurious) features S, the graph's latent representation Z, and the graph's label Y. The arrows indicate causation from one variable to another. The complete details about the derivation of the SCM can be found in Sui et al. (2022) and Zhao et al. (2024).

Given an input graph G, we define the *Causal Information Flow* as the Information Flow between the graph's causal features C and the graph's label Y. That is, $I(C \to Y)$ as in Equation (14). In practice, C is unobserved but can be sieved via a CGNN's causal channel. Fixed a CGNN, C can be extracted via the causal disentangling map g_ω^c, defined in Equation (7). Further, the causal disentangled representation h_c can be "pulled back" via the causal attention coefficients α_c^{n}, defined in Equation (3), to yield G's causal sub-structure G_c. Abusing notation, $G_c = \alpha_c^{\mathrm{n}} \odot G$. Figure 4 provides some examples of extracted causal sub-structures.

5.3 A LOSS FOR CGNN TRAINING DERIVED FROM THE CAUSAL INFORMATION FLOW

Out-of-Distribution graph-level classification (Problem 1) is a highly challenging task as the feature-label correlations present in the training set do not hold at test time. Only the causal (invariant) relations hold at test time. Hence, given an input graph $G \in \mathbb{G}$, we must learn a CGNN capable of predicting G's label by maximally disentangling its causal from its spurious features. In other words, we must learn a CGNN that maximizes the Causal Information Flow $I(C \to Y)$. Generally, computing information flows is challenging due to the presence of interventional conditioning (i.e., $do(\cdot)$ operations). Nonetheless, as we show in Theorem 5.2, $I(C \to Y)$ admits a tractable supervised lower bound.

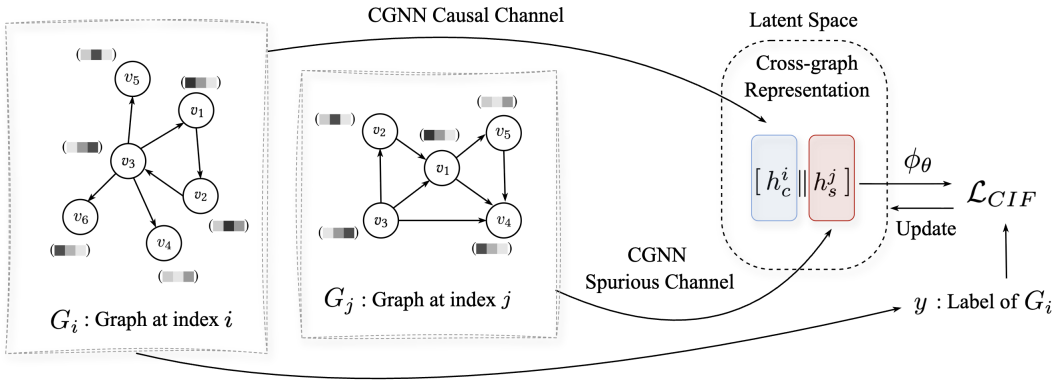

Figure 3: Causal Information Flow-based CGNN training.

Theorem 5.2 *The Causal Information Flow $I(C \to Y)$ admits a tractable supervised lower bound*

$$I(C \to Y) \geq \mathbb{E}_{G,G',y}\left[\log\left(\left[\phi_\theta\left([\, g_\omega^c(G) \,\|\, g_\omega^s(G')\,]\right)_y\right]\right)\right] + H(Y), \tag{15}$$

where ϕ_θ is the CGNN prediction head defined in Equation (8), g_ω^c and g_ω^s are the CGNN causal and spurious attention-based disentangling maps defined in Equation (7), and $H(\cdot)$ is the entropy.

Proof 5.1 (Sketch) *The bound follows from the application of the back-door adjustment formula and Jensen's inequality. Refer to Appendix A.4 for the complete argument.*

Our optimization is on the CGNN parameters $\theta \in \Theta$ and $\omega \in \Omega$. Since the entropy $H(Y)$ does not depend on neither, we can ignore it as it is constant. That is, we seek

$$\arg\max_{\theta\in\Theta,\omega\in\Omega} \mathbb{E}_{G,G',y}\left[\log\left(\phi_\theta\left([\, g_\omega^c(G) \,\|\, g_\omega^s(G')\,]\right)_y\right)\right]. \tag{16}$$

Based on Equation (16), we define the loss function

$$\mathcal{L}_{CIF} = -\frac{1}{N_G^2}\sum_{i=1}^{N_G}\sum_{j=1}^{N_G}\sum_{y} y \cdot \log(\phi_\theta\left([\, h_i^c \,\|\, h_j^s\,]\right)), \tag{17}$$

where y denotes the label of the graph G_i, $h_c^i = g_\omega^c(G_i)$ and $h_s^j = g_\omega^s(G_j)$ denote the disentangled causal and spurious representations of the graphs G_i and G_j, and ϕ_θ is the CGNN prediction head. The CIF loss is, in essence, a causal counterpart of the cross-entropy loss for graph-level classification. For each input graph G_i, its spurious representation h_s^i is exchanged with those of every other graph input graph G_j. The label y of G_i must then be recovered from the concatenated *cross-graph* representation $z = [h_c^i || h_s^j]$. Notably, the invariance to spurious features is not imposed through auxiliary terms but emerges from the objective itself. Figure 3 schematically depicts $\mathcal{L}_{CIF}$.

6 Experimental Evaluation

6.1 Experimental Setup

We conducted our experiments on a server powered with an NVIDIA A100-PCIE-40GB GPU using PyTorch 2.3 (Paszke et al., 2019) and PyG 2.5 (Fey & Lenssen, 2019) as computing infrastructure.

6.1.1 Compared Methods

We compared CIF against 2 foundational CGNN training paradigms: CAL (Sui et al., 2022) and DIR (Wu et al., 2022), and against 2 state-of-the-art CGNN training paradigms: ICL (Zhao et al., 2024) and ACE (Chen et al., 2025). The CGNN model class is graph-encoder agnostic as the GNN operator in Equation (1) can be materialized through any graph encoder. Hence, we instantiate it via 3 classic MPNNs: GCN (Kipf & Welling, 2016), GAT (Petar et al., 2018), GIN (Xu et al., 2018), and 3 state-of-the-art GTs: GraphGPS (Rampášek et al., 2022), GrokFormer (Pang et al., 2025), and DualFormer (Zhuo et al., 2025). This leads to a total of 30 unique method-backbone combinations.

Table 1: Characteristics of the datasets used in the experimental evaluation.

Name	Domain	# of graphs	Av. nodes	Av. edges	# node features	# edge features	# classes
SYN-β	Synthetic	8,000	238.75	385.25	10	–	4
SpMotif-β	Synthetic	12,000	53.99	38.46	4	1	3
MNIST-75sp-η	CV	35,000	70.55	295.14	5	1	10
Molhiv	Chemistry	41,127	25.51	27.46	9	3	2
Graph-SST2	NLP	70,042	10.19	9.19	768	1	2

6.1.2 DATASETS

We asses the effectiveness of CIF on 14 OOD graph-level classification benchmark datasets from 4 domains and 5 sources, covering synthetic and real datasets with synthetic shifts, and real datasets with real shifts. Their characteristics are summarized in Table 1. *SYN-β* (Sui et al., 2022) is a synthetic graph dataset family with a synthetic structural distribution shift: The distribution of base graph topologies (Tree vs. Barabasi-Albert) changes between train and test time across five bias levels $\beta \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$, where $\beta = 0.5$ refers to the unbiased scenario. That is, the bias level grows as $|\beta - 0.5|$. *SpMotif-β* (Zhao et al., 2024) is also a synthetic graph dataset family with a synthetic structural distribution shift that is constructed in a similar fashion. *MNIST-75sp-η* (Knyazev et al., 2019) (the OOD version of MNIST-sp (Monti et al., 2017)) is a real dataset of MNIST images converted to superpixel graphs with a synthetic node-feature distribution shift: The RGB color values of the superpixels are corrupted at test time across four noise levels $\eta \in \{0.2, 0.4, 0.6, 0.8\}$, where higher values correspond to stronger color corruption. Last, Molhiv (Hu et al., 2020) and Graph-SST2 (Yuan et al., 2022), are two real datasets with real distribution shifts.

6.1.3 METRICS

Graph-level classification performance is measured with AUPRC, the area under the precision recall curve (Boyd et al., 2013) and AUROC, the area under the receiver operating characteristic curve (Bradley, 1997). Both curves allow the evaluation of classifiers over a range of decision thresholds. For a global score of the models' performance, we additionally report mean AUROC and mean AUNPRC, averaged across datasets. AUNPRC, the Normalized AUPRC (Boyd et al., 2012), accounts for class skew so it can be meaningfully averaged over multiple datasets.

Invariant graph rationale discovery is assessed on the datasets with known ground-truth causal motifs. We resort on Precision@k (Li et al., 2023) and node-level attention AUROC (Bradley, 1997). Here, k denotes the number of nodes of the causal motif, and node-level labels are 1 if the node belongs to the motif, and 0 otherwise. For a global score of the models' performance, we additionally report mean Precision@k and mean node-level attention AUROC, averaged across datasets.

6.2 RESULTS

Table 2 presents the performance of the compared methods on both tasks. To keep the comparison among all 30 method-backbone combinations tractable, the results are aggregated across datasets. For completeness, the disaggregated results are reported in Tables 4, 5, and 6 in Appendix B. Overall, as can be observed in Table 2, CIF leads to competitive CGNN performance. CIF achieves 13 wins across both tasks and both GNN backbone families. Of these, 5 are outright, more than any other CGNN training paradigm achieves. We now discuss the per-task results in detail.

Graph-Level Classification Table 2a presents the performance of the compared methods on the graph-level classification task. We make two key observations. First, we find that the CIF training paradigm leads to the most first-place appearances in CGNN classification performance. CIF-based training results in 7 first-place appearances, out of which 3 are outright. The next best methods, ICL and ACE, feature 6 first place appearances but none of them are outright. Further, CAL comes in first place 4 times and DIR only once, being the lowest performing methods. Second, we observe that the backbone choice has an impact on CGNN classification performance under every training paradigm. GraphGPS leads to the best results followed by GIN. However, GT backbones do not generally lead to better performance than MPNN ones. For instance, GCN achieves better results than GrokFormer.

Table 2: Experimental results, aggregated across datasets.

(a) Graph-Level Classification. We report mean AUROC and mean AUNPRC aggregated across datasets. The corresponding disaggregated results can be found in Table 4 and Table 5 in Appendix B.

Backbone	GCN		GAT		GIN		GraphGPS		GrokFormer		DualFormer	
Method	AUROC	AUNPRC	AUROC	AUNPRC	AUROC	AUNPRC	AUROC	AUNPRC	AUROC	AUNPRC	AUROC	AUNPRC
DIR	0.77	**0.53**	0.75	0.51	0.79	0.57	0.80	0.59	0.68	0.36	0.75	0.51
CAL	**0.78**	**0.53**	0.75	0.51	**0.80**	**0.58**	0.80	0.59	0.67	0.34	0.75	0.51
ICL	0.77	**0.53**	0.75	0.51	**0.80**	0.57	**0.81**	**0.60**	0.57	0.20	**0.76**	**0.52**
ACE	0.77	**0.53**	0.75	**0.52**	**0.80**	0.57	**0.81**	0.59	0.67	0.35	**0.76**	**0.52**
CIF (us)	0.77	**0.53**	**0.76**	**0.52**	0.79	0.57	**0.81**	**0.60**	**0.69**	**0.37**	0.75	0.51

(b) Invariant Graph Rationale Discovery. We report mean Precision@k and mean node-level attention AUROC aggregated across datasets. The disaggregated results can be found in Table 6 in Appendix B.

Backbone	GCN		GAT		GIN		GraphGPS		GrokFormer		DualFormer	
Method	Prec@k	AUROC	Prec@k	AUROC	Prec@k	AUROC	Prec@k	AUROC	Prec@k	AUROC	Prec@k	AUROC
DIR	**0.68**	**0.77**	0.69	0.81	0.75	0.87	0.48	0.66	0.09	0.41	**0.66**	**0.76**
CAL	**0.68**	0.76	0.69	0.80	0.75	0.87	0.52	0.72	**0.14**	**0.46**	**0.66**	**0.76**
ICL	**0.68**	0.76	0.69	0.80	**0.76**	**0.88**	**0.74**	**0.86**	0.02	0.34	**0.66**	**0.76**
ACE	**0.68**	0.76	0.69	0.80	0.75	0.87	0.50	0.70	0.13	0.43	**0.66**	**0.76**
CIF (us)	**0.68**	**0.77**	**0.71**	**0.82**	0.73	0.86	0.50	0.67	0.12	0.44	**0.66**	**0.76**

Invariant Graph Rationale Discovery Table 2b presents the performance of the compared methods on the invariant graph rationale discovery task. We draw two main conclusions. First, despite not being the undisputed champion, CIF still exhibits competitive invariant graph rationale discovery capabilities. ICL is the strongest method with 7 first places (4 outright), closely followed by CIF with 6 first places (2 outright). CAL also proves to be competitive with 5 first place appearances, 2 of which are outright. DIR and ACE are the lowest performing methods on the task with only 4 and 3 first places, respectively. Second, we observe that the backbone choice has a significant effect on the invariant rationale discovery capabilities of the CGNNs under every training paradigm. In specific, the MPNN backbones are decisively better than the GT backbones, specifically GIN and GAT. The success of the local message passing architectures aligns with the local nature of invariant graph rationales. For illustration purposes, we provide two examples of extracted invariant graph rationales in Figure 4. The ground truth motif is a C_5 and the grayscale indicates node-level causal attention, with darker nodes receiving higher causal attention coefficients α_c^{n}.

6.3 Ablation Study

The loss function $\mathcal{L}_{\text{CIF}}$ in Equation (17) is defined via the lower bound established in Theorem 5.2. Deriving this bound involves the application of Jensen's inequality, trading off optimality for a numerically stable objective that imposes a stronger condition over ϕ_θ. Namely, incorrect predictions are penalized for every causal-spurious cross-graph swap, rather than only for the average predic-

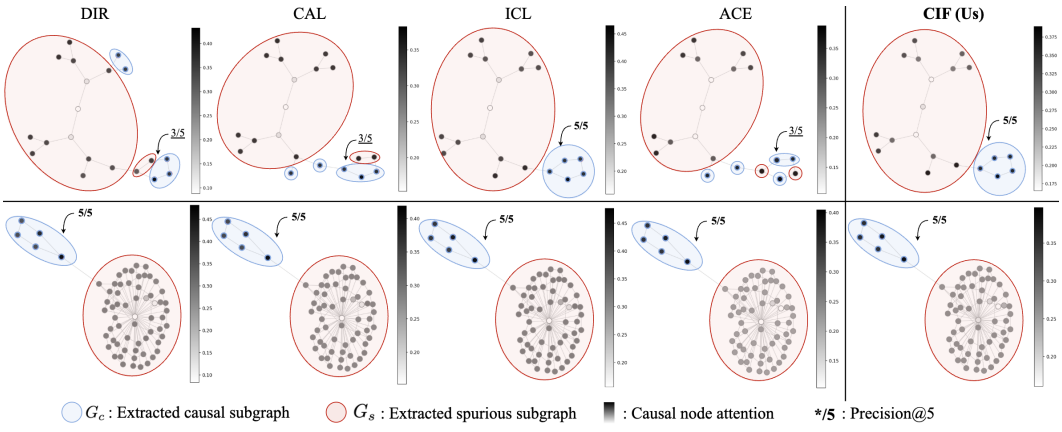

Figure 4: Extracted Invariant Graph Rationales G_c (blue) on SpMotif $b = 0.9$, GIN backbone.

Table 3: Ablation study. We report mean AUROC, mean AUNPRC, mean Precision@k and mean node-level attention AUROC, aggregated across datasets. The disaggregated results are provided in Tables 7, 8, and 9 in Appendix B. The aggregated results exclude missing runs.

Task	**Classification**						**Graph Rationale Discovery**			
Metric	AUROC			AUNPRC			Prec@k		AUROC	
Backbone \ Model	**CIF**	CIF$_{\leq}$	CE	**CIF**	CIF$_{\leq}$	CE	**CIF**	CIF$_{\leq}$	**CIF**	CIF$_{\leq}$
GCN	**0.77**	0.76	0.74	**0.53**	0.52	0.48	**0.68**	**0.68**	**0.77**	**0.77**
GAT	**0.76**	0.75	0.73	**0.52**	0.50	0.47	**0.71**	**0.71**	**0.82**	0.81
GIN	**0.79**	0.78	**0.79**	**0.57**	0.55	0.55	**0.73**	**0.73**	**0.86**	**0.86**
GraphGPS	**0.81**	0.80	0.79	**0.60**	0.58	0.59	0.50	**0.55**	0.67	**0.71**
GrokFormer	**0.69**	0.62	0.66	**0.37**	0.25	0.32	**0.12**	0.04	**0.44**	0.36
DualFormer	**0.75**	0.71	0.67	**0.51**	0.45	0.35	**0.66**	0.65	0.76	**0.78**

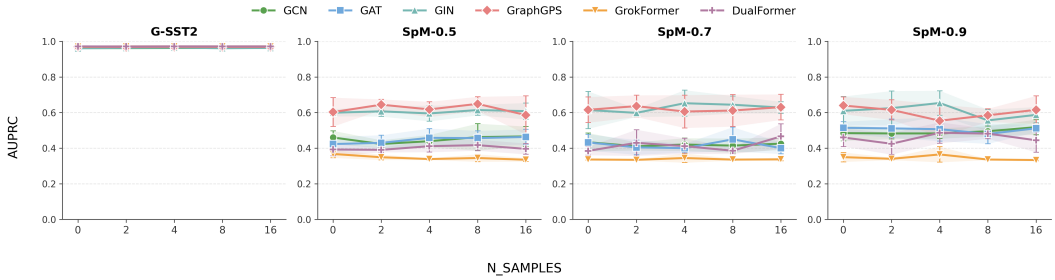

Figure 5: Sensitivity Analysis. MC approximation sharpness vs AUPRC (mean $\pm$ stdev).

tion over spurious representations for each fixed causal representation. More details can be found in Appendix A.4. To evaluate the effectiveness of this choice, we design an ablation model where we undo the application of Jensen's inequality from $\mathcal{L}_{\text{CIF}}$. We denote this ablation model by CIF$_{\leq}$. Additionally, we report the classification performance of standard cross-entropy (CE) training as a non-causal baseline. As can be appreciated in Table 3, CIF leads to better results than CIF$_{\leq}$ in classification and both have a comparable performance in rationale discovery. This suggests that the benefits derived from applying the Jensen's inequality outweigh the potential risks resulting from the introduced optimality gap. Further, CIF generally leads to better classification results than CE, supporting the benefit of causal-based learning over correlation-based learning on OOD graph-level classification.

6.4 Sensitivity Analysis

The CGNN model class introduced in Section 4 is deterministic. Hence, the integrals over the spurious features s and the latent representation z that appear when deriving $\mathcal{L}_{\text{CIF}}$, collapse into a single term. Refer to Appendix A.3 for the details. To assess whether this collapse sacrifices classification performance, we compare CIF against a stochastic variant that retains these integrals and approximates them via Monte Carlo sampling, perturbing $s^{(jl)} = g^s_\omega(G_j) + \epsilon^{(l)}$ and $z^{(ijln)} = [c^{(i)}||s^{(jl)}] + \epsilon^{(n)}$ with $\epsilon^{(l)}, \epsilon^{(n)} \sim \mathcal{N}(0, I)$ through the reparameterization trick. We vary $N_s = N_z \in \{2, 4, 8, 16\}$ while keeping the batch size N_G fixed. The deterministic version of CIF is denoted by $N_s = N_z = 0$ in Figure 5. As Figure 5 shows, no number of samples improves classification performance. Hence, both sampling steps can be effectively omitted without sacrificing classification performance.

7 Conclusion

In this work we introduce CIF, to the best of our knowledge, the first causal-by-design CGNN training paradigm based on the optimization of a single optimization objective, in essence, a causal counterpart of the cross-entropy loss for graph-level classification. Our extensive experimental evaluation shows that CIF leads to competitive CGNN performance while optimizing a single objective.

AI USE STATEMENT

In this work, we have not used generative AI tools for any task that requires disclosure. With respect to the tasks with recommended disclosure: Our experimental evaluation is very extensive, 14 datasets, 6 methods and 6 backbones. Each such method-backbone combination is run 5 times with different seeds, producing a unique result file for each. Manually parsing all those results becomes infeasible. For this matter, we used generative AI tools (Claude code with Claude Opus 4.8 backbone) to generate scripts that automatically read all the result files and parse them into the latex result tables. The same tool was use for file and variable name refactoring when preparing the final version of the code repository. LLM-generated code was verified and tested for correctness by the main author. Additionally, we used generative AI tools (Claude Opus 4.8) for editing section 4 of the paper to better organize the presentation of the formulas with the aim of improving readability of that section. We have reviewed all AI-assisted work. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

With respect to our theoretical findings: We include the proofs for all the theorems in Appendix A. With respect to the experimental evaluation: We make our code and datasets publicly available at https://anonymous.4open.science/r/CIF-51E7. The link additionally includes the code for all of the considered baselines as well as all the data processing files. All of our results can be reproduced by running `./run_experiments.sh`. Last, we make a note about one of the used datasets: The MNIST-75sp dataset was originally introduced in (Knyazev et al., 2019). However, the exact version of the dataset and the processing that we use is that of (Zhao et al., 2024), available at https://github.com/DataLab-atom/ICL/blob/main/datasets.py. This includes sampling 20K training graphs, and 5K validation graphs as detailed in https://github.com/Wuyxin/DIR-GNN/blob/main/datasets/mnistsp_dataset.py (Wu et al., 2022). This explains why there are 35K graphs rather than 70K. The raw files are available at https://drive.google.com/drive/folders/1Prc-n9Nr8-5z-xphdRScftKKIxU4Olzh (Wu et al., 2022). Moreover, the default noise level is 0.4. We further create 3 more versions of the dataset with noise levels $0.2, 0.6, 0.8$. The corresponding data processing file along with that of all of the other datasets we used can be found under `data_scripts` on our code repository.

REFERENCES

Carlo Abate, Sergio Decherchi, and Andrea Cavalli. Graph neural networks for conditional de novo drug design. *Wiley Interdisciplinary Reviews: Computational Molecular Science*, 13(4):e1651, 2023.

Weizhi An, Wenliang Zhong, Feng Jiang, Hehuan Ma, and Junzhou Huang. Causal subgraphs and information bottlenecks: Redefining ood robustness in graph neural networks. In *European Conference on Computer Vision*, pp. 473–489. Springer, 2024.

Nihat Ay and Daniel Polani. Information flows in causal networks. *Advances in complex systems*, 11(01):17–41, 2008.

Kendrick Boyd, Vitor Santos Costa, Jesse Davis, and C David Page. Unachievable region in precision-recall space and its effect on empirical evaluation. In *Proceedings of the... International Conference on Machine Learning. International Conference on Machine Learning*, volume 2012, pp. 349, 2012.

Kendrick Boyd, Kevin H Eng, and C David Page. Area under the precision-recall curve: point estimates and confidence intervals. In *Joint European conference on machine learning and knowledge discovery in databases*, pp. 451–466. Springer, 2013.

Andrew P Bradley. The use of the area under the roc curve in the evaluation of machine learning algorithms. *Pattern recognition*, 30(7):1145–1159, 1997.

Jiali Chen, Tiefeng Ma, and Liwen Xu. Revolutionizing graph classification generalization with an adaptive causality-enhanced framework. *Neurocomputing*, 631:129716, 2025.

Yongqiang Chen, Yonggang Zhang, Yatao Bian, Han Yang, MA Kaili, Binghui Xie, Tongliang Liu, Bo Han, and James Cheng. Learning causally invariant representations for out-of-distribution generalization on graphs. *Advances in Neural Information Processing Systems*, 35:22131–22148, 2022.

Yongqiang Chen, Yatao Bian, Bo Han, and James Cheng. How interpretable are interpretable graph neural networks? In *Proceedings of the 41st International Conference on Machine Learning*, pp. 6413–6456, 2024.

Mark Cheung and José MF Moura. Graph neural networks for covid-19 drug discovery. In *2020 IEEE International Conference on Big Data (Big Data)*, pp. 5646–5648. IEEE, 2020.

Enyan Dai and Suhang Wang. Towards self-explainable graph neural network. In *Proceedings of the 30th ACM international conference on information & knowledge management*, pp. 302–311, 2021.

Jiale Deng and Yanyan Shen. Self-interpretable graph learning with sufficient and necessary explanations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 11749–11756, 2024.

Shaohua Fan, Xiao Wang, Yanhu Mo, Chuan Shi, and Jian Tang. Debiasing graph neural networks via learning disentangled causal substructure. *Advances in Neural Information Processing Systems*, 35:24934–24946, 2022.

Chenchen Feng, Yu He, Shiyang Wen, Guojun Liu, Liang Wang, Jian Xu, and Bo Zheng. Dcgnn: Decoupled graph neural networks for improving and accelerating large-scale e-commerce retrieval. In *Companion Proceedings of the Web Conference 2022*, pp. 32–40, 2022.

Matthias Fey and Jan Eric Lenssen. Fast graph representation learning with pytorch geometric. *arXiv preprint arXiv:1903.02428*, 2019.

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

Weihua Hu, Matthias Fey, Marinka Zitnik, Yuxiao Dong, Hongyu Ren, Bowen Liu, Michele Catasta, and Jure Leskovec. Open graph benchmark: Datasets for machine learning on graphs. *Advances in neural information processing systems*, 33:22118–22133, 2020.

Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

Boris Knyazev, Graham W Taylor, and Mohamed Amer. Understanding attention and generalization in graph neural networks. *Advances in neural information processing systems*, 32, 2019.

Nguyen Quoc Khanh Le. Predicting emerging drug interactions using gnns. *Nature Computational Science*, 3(12):1007–1008, 2023.

Xujia Li, Yuan Li, Xueying Mo, Hebing Xiao, Yanyan Shen, and Lei Chen. Diga: Guided diffusion model for graph recovery in anti-money laundering. In *SIGKDD*, pp. 4404–4413, 2023.

Junhong Lin, Xiaojie Guo, Yada Zhu, Samuel Mitchell, Erik Altman, and Julian Shun. Fraudgt: A simple, effective, and efficient graph transformer for financial fraud detection. In *Proceedings of the 5th ACM International Conference on AI in Finance*, pp. 292–300, 2024.

Wanyu Lin, Hao Lan, Hao Wang, and Baochun Li. Orphicx: A causality-inspired latent variable model for interpreting graph neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13729–13738, 2022.

Yixin Liu, Kaize Ding, Qinghua Lu, Fuyi Li, Leo Yu Zhang, and Shirui Pan. Towards self-interpretable graph-level anomaly detection. *Advances in Neural Information Processing Systems*, 36:8975–8987, 2023.

Siqi Miao, Mia Liu, and Pan Li. Interpretable and generalizable graph learning via stochastic attention mechanism. In *International Conference on Machine Learning*, pp. 15524–15543. PMLR, 2022.

Federico Monti, Davide Boscaini, Jonathan Masci, Emanuele Rodola, Jan Svoboda, and Michael M Bronstein. Geometric deep learning on graphs and manifolds using mixture model cnns. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5115–5124, 2017.

Soroor Motie and Bijan Raahemi. Financial fraud detection using graph neural networks: A systematic review. *Expert Systems with Applications*, 240:122156, 2024.

Xichuan Niu, Bofang Li, Chenliang Li, Rong Xiao, Haochuan Sun, Hongbo Deng, and Zhenzhong Chen. A dual heterogeneous graph attention network to improve long-tail performance for shop search in e-commerce. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 3405–3415, 2020.

Guansong Pang, Hezhe Qiao, Hui Yan, et al. Grokformer: Graph fourier kolmogorov-arnold transformers. In *Forty-second International Conference on Machine Learning*, 2025.

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.

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

Veličković Petar, Cucurull Guillem, Casanova Arantxa, Romero Adriana, Lio Pietro, and B Yoshua. Graph attention networks. In *International conference on learning representations*, volume 8, 2018.

Ladislav Rampášek, Michael Galkin, Vijay Prakash Dwivedi, Anh Tuan Luu, Guy Wolf, and Dominique Beaini. Recipe for a general, powerful, scalable graph transformer. *Advances in Neural Information Processing Systems*, 35:14501–14515, 2022.

Ramon Rico, Ioana Hulpus, Stan Leisink, Boyang Zhao, and Yannis Velegrakis. Translating latent representations for money laundering detection. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 2026.

Yunchong Song, Chenghu Zhou, Xinbing Wang, and Zhouhan Lin. Ordered gnn: Ordering message passing to deal with heterophily and over-smoothing. In *The Eleventh International Conference on Learning Representations*, 2023.

Yongduo Sui, Xiang Wang, Jiancan Wu, Min Lin, Xiangnan He, and Tat-Seng Chua. Causal attention for interpretable and generalizable graph classification. In *Proceedings of the 28th ACM SIGKDD conference on knowledge discovery and data mining*, pp. 1696–1705, 2022.

Qiaoyu Tan, Ninghao Liu, and Xia Hu. Deep representation learning for social network analysis. *Frontiers in big Data*, 2:2, 2019.

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.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. *arXiv preprint arXiv:1710.10903*, 2017.

Ying-Xin Wu, Xiang Wang, An Zhang, Xiangnan He, and Tat seng Chua. Discovering invariant rationales for graph neural networks. In *ICLR*, 2022.

Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and Philip S Yu. A comprehensive survey on graph neural networks. *IEEE transactions on neural networks and learning systems*, 32(1):4–24, 2020.

Wenwen Xia, Yuchen Li, Jun Wu, and Shenghong Li. Deepis: Susceptibility estimation on social networks. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*, pp. 761–769, 2021.

Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? *arXiv preprint arXiv:1810.00826*, 2018.

Guipeng Xv, Chen Lin, Wanxian Guan, Jinping Gou, Xubin Li, Hongbo Deng, Jian Xu, and Bo Zheng. E-commerce search via content collaborative graph neural network. In *Proceedings of the 29th ACM SIGKDD conference on knowledge discovery and data mining*, pp. 2885–2897, 2023.

Junchi Yu, Tingyang Xu, Yu Rong, Yatao Bian, Junzhou Huang, and Ran He. Graph information bottleneck for subgraph recognition. In *International Conference on Learning Representations*, 2021.

Hao Yuan, Haiyang Yu, Shurui Gui, and Shuiwang Ji. Explainability in graph neural networks: A taxonomic survey. *IEEE transactions on pattern analysis and machine intelligence*, 45(5): 5782–5799, 2022.

Yanfu Zhang, Shangqian Gao, Jian Pei, and Heng Huang. Improving social network embedding via new second-order continuous graph neural networks. In *Proceedings of the 28th ACM SIGKDD conference on knowledge discovery and data mining*, pp. 2515–2523, 2022.

Zhe Zhao, Pengkun Wang, Haibin Wen, Yudong Zhang, Zhengyang Zhou, and Yang Wang. A twist for graph classification: Optimizing causal information flow in graph neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 17042–17050, 2024.

Jiaming Zhuo, Yuwei Liu, Yintong Lu, Ziyi Ma, Kun Fu, Chuan Wang, Yuanfang Guo, Zhen Wang, Xiaochun Cao, and Liang Yang. Dualformer: Dual graph transformer. In *The Thirteenth International Conference on Learning Representations*, 2025.