# Causal-Based Supervision of Attention in Graph Neural Network: A Better and Simpler Choice towards Powerful Attention

**Hongjun Wang** [1] , **Jiyuan Chen** [1] , **Lun Du** [2] , **Qiang Fu** [2] , **Shi Han** [2] , **Xuan Song** [1]
[1]Southern University of Science and Technology, Shenzhen, China
[2]Microsoft Research Asia, Beijing, China
{wanghj2020, chenjy6, songx}@mail.sustech.edu.cn,
{lun.du, qifu, shihan}@microsoft.com

## Abstract

Recent years have witnessed the great potential of attention mechanism in graph representation learning. However, while variants of attention-based GNNs are setting new benchmarks for numerous real-world datasets, recent works have pointed out that their induced attentions are less robust and generalizable against noisy graphs due to lack of direct supervision. In this paper, we present a new framework which utilizes the tool of causality to provide a powerful supervision signal for the learning process of attention functions. Specifically, we estimate the direct causal effect of attention to the final prediction, and then maximize such effect to guide attention attending to more meaningful neighbors. Our method can serve as a plug-and-play module for any canonical attention-based GNNs in an end-to-end fashion. Extensive experiments on a wide range of benchmark datasets illustrated that, by directly supervising attention functions, the model is able to converge faster with a clearer decision boundary, and thus yields better performances.

## 1 Introduction

Graph-structured data is widely used in real-world domains, such as social networks [Zhang and Chen, 2018], recommender systems [Wu *et al.*, 2022b], and biological molecules [Gilmer *et al.*, 2017]. The non-euclidean nature of graphs has inspired a new type of machine learning model, Graph Neural Networks (GNNs) [Kipf and Welling, 2016; Defferrard *et al.*, 2016; Du *et al.*, 2022]. Generally, GNN iteratively updates features of the center node by aggregating those of its neighbors and has achieved remarkable success across various graph analytical tasks. However, the aggregation of features between unrelated nodes has long been an obstacle for GNN, keeping it from further improvement.

Recently, Graph Attention Network (GAT) [Veličković *et al.*, 2017] pioneered the adoption of the attention mechanism, a well-established method with proven effectiveness in deep learning [Vaswani *et al.*, 2017], into the neighborhood aggregation process of GNNs to alleviate the issue. The key concept behind GAT is to adaptively assign importance to each neighbor during the aggregation process. Its simplicity and effectiveness have made it the most widely used variant of GNN. Following this line, a myriad of attention-based GNNs have been proposed and have achieved state-of-the-art performance in various tasks [Sun *et al.*, 2021; Zhang *et al.*, 2022; Ying *et al.*, 2021; Brody *et al.*, 2021].

Nevertheless, despite the widespread use and satisfying results, in the past several years, researchers began to rethink if the learned attention functions are truly effective [Kim and Oh, 2022; Wang *et al.*, 2019; Wu *et al.*, 2022a; Knyazev *et al.*, 2019; Liu *et al.*, 2021; Wang *et al.*, 2021]. As we know, most existing attention-based GNNs learn the attention function in a weakly-supervised manner, where the attention modules are simply supervised by the final loss function, without a powerful supervising signal to guide the training process. And the lack of direct supervision on attention might be a potential cause of a less robust and generalizable attention function against real-world noisy graphs [Sui *et al.*, 2022; Knyazev *et al.*, 2019; Li *et al.*, 2022; Wang *et al.*, 2019; Ye and Ji, 2021; Huang *et al.*, 2023]. To address this problem, existing work enhances the quality of attention through auxiliary regularization terms (supervision). However, concerns have been raised that these methods often rely heavily on human-specified prior assumptions about a specific task, which limits their generalizability [You *et al.*, 2020; Wu *et al.*, 2022a]. Additionally, the auxiliary regularization is formulated independently of the primary prediction task, which may disrupt the original optimization target and cause the model to “switch” to a different objective function during training [Kim and Oh, 2022; You *et al.*, 2020].

Recently, causal inference [Pearl, 2009] has attracted many researchers in the field of GNNs by utilizing structural causal model (SCM) [Cinelli *et al.*, 2019] to handle distribution shift [Zhao *et al.*, 2022] and shortcut learning [Feng *et al.*, 2021]. In this paper, we argue that the tool of causal inference has also shed light on a promising avenue that could supervise and improve the quality of GNN’s attention directly, while in the meantime we will not make any assumptions about specific tasks or models, and the supervision signal for attention implicitly aligns well with the primary task. Before going any deeper, we first provide a general schema for the SCM of attention-based GNNs in Figure 1, which uses nodes to represent variables and edges to indicate causal relations between variables. As we can see, after a high-level abstraction, there are only three key factors in SCM, including the node

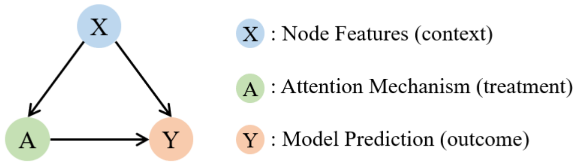



Figure 1: Structural causal model of attention-based GNNs

features $X$, attention maps $A$, and the model's final prediction $Y$. Note that in causal language, $X$, $A$, and $Y$ also denotes the context, treatment, and outcome respectively. For edges in SCM, the link $X \rightarrow A$ represents that the attention generation relies on the node's features (i.e., context decides treatment). And links $(X, A) \rightarrow Y$ indicate that the model's final prediction is based on both the node's features $X$ and the attention $A$ (i.e., the final outcome is jointly determined by both context and treatment).

In order to provide a direct supervision signal and further enhance the learning of attention functions, the first step would be finding a way to measure the quality of attention (i.e., quantifying what to improve). Since there are no unified criteria on the way of measurement, researchers usually propose their own solution according to the tasks they are facing, and this is very likely to introduce the unfavorable human-intervened prior assumptions [You *et al.*, 2020]. For example, CGAT [Wang *et al.*, 2019] believes that better attention should focus more on one-hop neighbors. While this assumption surely works on homophilous graphs, it will suffer from huge performance degradation in heterophilous scenarios. Our method differs a lot from existing work in that we introduce SCM to effectively decouple the direct causal effect of attention on the final prediction (i.e., link $A \rightarrow Y$), and use such causal effect as a measurement for the quality of attention. In this way, it is the model and data that decide if the attention works well during training instead of human-predefined rules. And this has been shown to be non-trivial in various machine learning fields because what might seem reasonable to a human might not be considered the same way by the model [Kumar *et al.*, 2010; Wang *et al.*, 2022b]. Another drawback of existing attention regularization methods, as previously mentioned, is the deviation from primary tasks. SuperGAT [Kim and Oh, 2022] uses link prediction to improve the attention quality for node classification, but as the author claims in the paper, there is an obvious trade-off between the two tasks. In this paper, we alleviate this problem by directly maximizing the causal effect of attention on the primary task (i.e., strengthening the causal relation $A \rightarrow Y$). Under mild conditions, we can deem the overall optimization is still towards the primary objective, except that we additionally provide a direct and powerful signal for the learning of attention in a fully-supervised manner.

In summary, this paper presents a **C**ausal **S**upervision for **A**ttention in graph neural networks (abbreviated as **CSA** in the following paragraphs). CSA has strong applicability because no human-intervened assumptions are made on the target models or training tasks. And the supervision of CSA can be easily and smoothly integrated into optimizing the primary task to performing end-to-end training. We list the main contributions in this paper as follows:

- We explore and provide a brand-new perspective to directly boost GNN's attention with the tool of causality. To the best of our knowledge, this is a promising direction that still remains unexplored.
- We propose CSA, a novel causal-based supervision framework for attention in GNNs, which can be formulated as a simple yet effective external plug-in for a wide range of models and tasks to improve their attention quality.
- We perform extensive experiments and analysis on CSA and the universal performance gain on standard benchmark datasets validates the effectiveness of our design.

## 2 Related Work

**Attention-based Graph Neural Networks.** Modeling pairwise importance between elements in graph-structured data dates back to interaction networks [Battaglia *et al.*, 2016; Hoshen, 2017] and relational networks [Santoro *et al.*, 2017]. Recently GAT [Veličković *et al.*, 2017] rose as one of the representative work of attention-based GNNs using self-attention [Vaswani *et al.*, 2017]. The remarkable success of GAT in multiple tasks has motivated many works focusing on integrating attention into GNN [Thekumparampil *et al.*, 2018; Zhang *et al.*, 2018; Wang *et al.*, 2022a; Zhang *et al.*, 2020; Gao and Ji, 2019; Hou *et al.*, 2022]. *Lee et al.* have also conducted a comprehensive survey [Lee *et al.*, 2019] on various types of attention used in GNNs.

**Causal Inference in Graph Neural Network.** Causality [Pearl, 2014] provides researchers new methodologies to design robust measurements, discover hidden causal structures and confront data biases. A myriad of studies has shown that incorporating causality is beneficial to graph neural network in various tasks. [Zhao *et al.*, 2022] makes use of counterfactual links to augment data for link prediction improvement. [Sui *et al.*, 2022] performs interventions on the representations of graph data to identify the causally attended subgraph for graph classification. [Feng *et al.*, 2021] on the other hand, applies causality to estimate the causal effect of node's local structure to assist node classification.

**Improving Attention in GAT.** There is a great number of work dedicated to improving attention learning in GAT. [Kim and Oh, 2020] enhances attention by exploiting two attention forms compatible with a self-supervised task to predict edges. [Brody *et al.*, 2021] introduces a simple fix by modifying the order of operations in GAT. [Wang *et al.*, 2019] develops an approach using constraint on the attention weights according to the class boundary and feature aggregation pattern. In addition, causality also plays a role in boosting the attention of GATs recently. [Wu *et al.*, 2022a] estimates the causal effect of edges by intervention and regularizes edges' attention weights according to their causal effects.

## 3 Preliminaries

We start by introducing the notations and formulations of graph neural networks and their attention variant. Let $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ represents a graph where $\mathcal{V} = \{v_i\}_{i=0}^{n}$ is the set of nodes and $\mathcal{E} \in \mathcal{V} \times \mathcal{V}$ is the set of edges. For each node $v \in \mathcal{V}$,

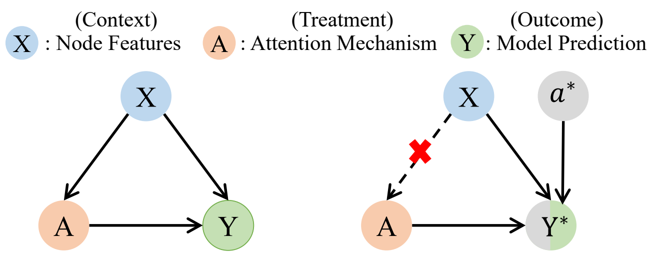



Figure 2: Deriving causal effects through counterfactual

it has its own neighbor set $N(v) = \{u \in \mathcal{V} \mid (v,u) \in \mathcal{E}\}$ and its initial feature vector $x_v^0 \in \mathbb{R}^{d^0}$, where $d^0$ is the original feature dimension. Generally, GNN follows the message-passing mechanism to perform feature updating, where each node's feature representation is updated by aggregating the representations of its neighbors and then combining the aggregated messages with its ego representation [Xu *et al.*, 2018]. Let $m_v^l \in \mathbb{R}^{d^l}$ and $x_v^l \in \mathbb{R}^{d^l}$ be the message vector and representation vector of node $v$ at layer $l$, we formally define the updating process of GNN as:

$$m_v^l = \text{AGGREGATE}\left(\left\{x_j^{(l-1)}, \forall j \in N(v)\right\}\right)$$
$$x_v^l = \text{COMBINE}\left(x_v^{l-1}, m_v^l\right),$$

where AGGREGATE and COMBINE are aggregation functions (e.g., mean, LSTM) and combination function (e.g., concatenation), respectively. The design of these two functions is what mostly distinguishes one type of GNN from the other. GAT [Veličković *et al.*, 2017] augments the normal aggregation with the introduction of self-attention. The core idea of self-attention in GAT is to learn a scoring function that computes an attention score for every node in $N(v)$ to indicate their relational importance to node $v$. In layer $l$, such process is defined by the following equation:

$$e\left(x_{v_i}^l, x_{v_j}^l\right) = \sigma\left((\mathbf{a}^l)^\top \cdot \left[W^l\, x_{v_i}^l \| W^l\, x_{v_j}^l\right]\right),$$

where $(\mathbf{a}^l, W^l)$, $\sigma$ are learnable matrices and activation function (e.g., LeakyReLU) respectively, and $\|$ denotes vector concatenation. The attention scores are then normalized across all neighbors $v_j \in N(v_i)$ using softmax to ensure consistency:

$$\alpha_{ij}^l = \frac{\exp\left(e\left(x_{v_i}^l, x_{v_j}^l\right)\right)}{\sum_{v_j \in N(v_i)} \exp\left(e\left(x_{v_i}^l, x_{v_j}^l\right)\right)}$$

Finally, GAT computes a weighted average of the features of the neighboring nodes as the new feature of $v_i$, which is demonstrated as follows:

$$x_{v_i}^{l+1} = \sigma\left(\sum\nolimits_{v_j \in N(v_i)} \alpha_{ij}^l W^l\, x_{v_j}^l\right).$$

# 4 Casual-based Supervision on GNN's Attention

In this section, we first introduce how the causal effect of attention can be derived from the structural causal model of attention-based GNNs. Specifically, this is done with the help of the widely used counterfactual analysis in causal reasoning. After that, with the obtained causal effects, we elaborate three candidate schemes to incorporate with the training of attention-based GNNs to improve their quality of attention.

## 4.1 Causal Effects of Attention

As previously mentioned, the first step towards improving attention lies in measuring the quality of existing attention. However, since deep learning models usually exhibit as black boxes, it is generally infeasible to directly assess their attention qualities. Existing works mainly address this issue by introducing human priors to build pre-defined rules for some specific models and tasks. Yet, it has been a long debate on whether human-made rules share consensus with deep learning models during training [Kumar *et al.*, 2010; Ribeiro *et al.*, 2016]. Fortunately, the recent rise of causal inference technology has offered effective tools to help us think beyond the black box and analyze causalities between model variables, which leads us to an alternative way to directly utilize the causal effect of attention to measure its quality. Since the obtained causal effects are mainly affected by the model itself, it is a more accurate and unbiased measurement of how well the attention actually learns.

We first give a brief review of the formulation of attention-based graph neural network in causal languages, as shown in Figure 2(a). The generated attention map $A$ is directly affected by node feature $X$. And the model prediction $Y$ is jointly determined by both $X$ and $A$. We denote the inferring process of the model as:

$$Y_{x,a} = Y(X = x, A = a), \tag{1}$$

which indicates that model will give value $Y_{x,a}$ if the value of $X$ and $A$ are set to $x$ and $a$ respectively. In order to pursue the attention's causal effect, we introduce the widely-used counterfactual analysis [Pearl, 2022] in causal reasoning.

The core idea of counterfactual causality lies in asking: given a certain data context (node feature $X$), what the outcome (model prediction $Y$) would have been if the treatment (attention map $A$) had not been the observed value? To answer the imaginary question, we have to manipulate the values of several variables to see the effect, and this is formally termed as *intervention* in causal inference literature, which can be denoted as $do(\cdot)$. In $do(\cdot)$ operation, we compulsively select a counterfactual value to replace the original factual value of the intervened variable. And once a variable is intervened, its all incoming links in the SCM will be cut off and its value is independently given, while other variables that are not affected still maintain the original value. In our case, for example, $do(A = a^*)$ means we demand the attention $A$ to take the non-real value $a^*$ (e.g., reverse/random attention) so that the link $X \rightarrow A$ is cut-off and $A$ is no longer be affected by its causal parent $X$. This process is illustrated in Figure 2(b) and the mathematical formulation is given as:

$$Y_{x,a^*} = Y(X = x, do(A = a^*)), \tag{2}$$

which indicates that after $do(\cdot)$ operation which changes the value of attention to be $a^*$, the output value of the model also changes to $Y_{x,a^*}$. Finally, let us consider a case where we

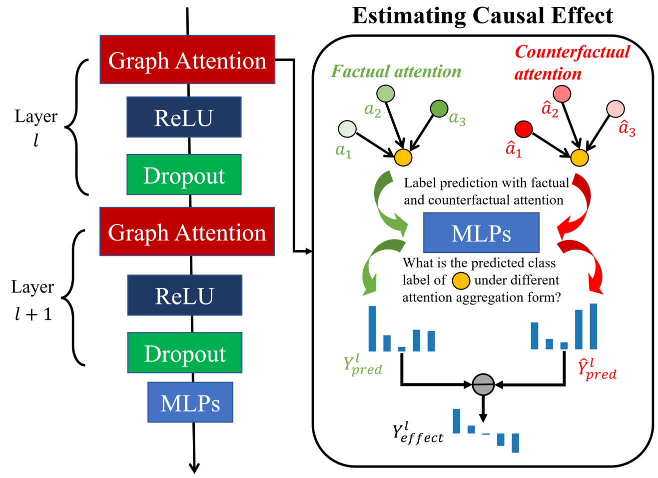



Figure 3: The schematic of CSA is shown above as a plug-in to graph attention methods. The $a$ and $\hat{a}$ indicate the factual and counterfactual attention values, respectively. We subtract the counterfactual classification results from the original classification to analyze the causal effects of learned attention (i.e., attention quality) and directly maximize them in the training process towards primary task.

assign a *dummy* value $\tilde{a}$ to the attention map so that for each ego node, all its neighbors share the same attention weights, the feature aggregation of the graph attention model will then degrade to an unweighted average. In this case, according to the theory of causal inferences [VanderWeele, 2016], the Total Direct Effect (TDE) of attention to model prediction can be obtained by computing the differences of model outcome $Y_{x,a}$ and $Y_{x,\tilde{a}}$, which is formulated as follows:

$$TDE = Y_{x,a} - Y_{x,\tilde{a}}. \tag{3}$$

It is worth noting that the induction of attention's causal effect does not exert any assumptions and constraints, which lays a solid foundation for our wide applicability on any graph attention models.

## 4.2 Supervision on Attention with Causal Effects

We have already demonstrated the derivation of attention's causal effect in the previous section. In this part, we will discuss how to utilize the obtained causal effect for attention quality improvement. Previous works that make use of an auxiliary task to regularize attention usually suffered from performance trade-off between the primary task and auxiliary task. In this work, we alleviate this problem by directly maximizing the causal effect of attention on the primary task. The overall schema of this part is shown in Figure 3.

Consider a simple case where we conduct the node classification task with a standard $L$-layer GAT. For each layer $l$, we have the node representations $X^{l-1} \in \mathbb{R}^{n \times d^{l-1}}$ from the previous layer as input. Then, we perform feature aggregation and updating with factual attention map $A^l$ to obtain the factual output feature $X^l = f(X^{l-1}, A^l)$. Similarly, when we intervene the attention maps of layer $l$ (e.g., assigning *dummy* values using $do(\cdot)$ operation), we can get a counterfactual output feature $\hat{X}^l$. We further employ a learnable matrix $\mathcal{W}^l \in \mathbb{R}^{c \times d^l}$ ($c$ denotes the number of classes) to get the node's factual predicted label $Y^l_{pred}$ and counterfactual predicted label $\hat{Y}^l_{pred}$ using the corresponding features from layer $l$. Therefore, the causal effect of attention at layer $l$ is obtained as: $Y^l_{pred} - \hat{Y}^l_{pred}$. To this end, we can use the causal effect as a supervision signal to explicitly guide the attention learning process. The new objective of the CSA-assisted GAT model can be formulated as:

$$\mathcal{L} = \sum_l \lambda_l \mathcal{L}_{ce}(Y^l_{\text{effect}}, y) + \mathcal{L}_{\text{others}}, \tag{4}$$

where $y$ is the ground-truth label, $\mathcal{L}_{ce}$ is the cross-entropy loss, $\lambda_l$ is the coefficient to balance training, and $\mathcal{L}_{\text{others}}$ represents the original objective such as standard classification loss. Note that Equation.(4) is a general form of CSA where we compute additional losses for each GAT layer to supervise attention directly. However in practice it is not necessary, and we found that simply selecting one or two layers is enough for CSA to bring satisfying performance improvement.

Moreover, since our aim is to boost the quality of attention, it is not necessary to estimate the correct causal effect of attention using *dummy* values. Instead, a strong counterfactual baseline might even be helpful for the attention quality improvement. We hereby further propose three heuristic counterfactual schemes and test them in our experiments. We note that the exact form of how counterfactual is achieved is not limited, and our goal here is just to set the ball rolling.

**Scheme I:** In the first scheme, we utilize the uniform distribution to generate the counterfactual attention map. Specifically, the counterfactual attention is produced by

$$\hat{a} \sim U(e, f), \tag{5}$$

where $e$ and $f$ are the lower and upper boundaries. In this case, the generated counterfactual could vary from very bad (i.e., mapping all unrelated neighbors) to very good (i.e., mapping all meaningful neighbors). This is a similar process to the *Randomized Controlled Trial* [Stolberg *et al.*, 2004] where all possible treatments are enumerated. We hope that maximizing causal effects computed over all possible treatments can lead to a robust improvement of attention.

**Scheme II:** Scheme I is easy and straightforward to apply. However due to its randomness, a possible concern is that if most of the generated counterfactual attentions are inferior to the factual one, then we will only have very small gradient on attention improvements. Therefore we are actually motivated to find "better" counterfactuals to spur the factual one to evolve. Heuristically, given that MLP is a strong baseline one several datasets (e.g., Texas, Cornell, and Wisconsin), we employ an identical mapping to generate the counterfactual attention, which only attends to the ego-node instead of neighbors. Specifically, the counterfactual attention map is equal to the identity matrix $I$:

$$\hat{a} \sim I \tag{6}$$

**Scheme III:** Our last schema can be considered as an extension of Schema II. Since the fast development of GAT family has introduced to us some variants that already outperform MLP in many datasets, using counterfactuals derived from

| | **Texas** | **Wisconsin** | **Actor** | **Squirrel** | **Chameleon** | **Cornell** | **Crocodile** |
|---|---|---|---|---|---|---|---|
| $\mathcal{H}(\mathcal{G})$ | $\mathbf{0.11}$ | $\mathbf{0.21}$ | $\mathbf{0.22}$ | $\mathbf{0.22}$ | $\mathbf{0.23}$ | $\mathbf{0.3}$ | $\mathbf{0.26}$ |
| **#Nodes** | 183 | 251 | 7,600 | 5,201 | 2,277 | 183 | 11,631 |
| **#Edges** | 295 | 466 | 191,506 | 198,493 | 31,421 | 280 | 899,756 |
| **#Classes** | 5 | 5 | 5 | 5 | 5 | 5 | 6 |
| **#Features** | 1703 | 1703 | 932 | 2089 | 2325 | 1703 | 500 |
| MLP | 81.32 ± 6.22 | 84.38 ± 5.34 | 36.09 ± 1.35 | 28.98 ± 1.32 | 46.21 ± 2.89 | 83.92 ± 5.88 | 54.35 ± 1.90 |
| SGCN | 56.41 ± 4.29 | 54.82 ± 3.63 | 30.50 ± 0.94 | 52.74 ± 1.58 | 60.89 ± 2.21 | 62.52 ± 5.10 | 51.80 ± 1.53 |
| GCN | 55.59 ± 5.96 | 53.48 ± 4.75 | 28.40 ± 0.88 | 53.98 ± 1.53 | 61.54 ± 2.59 | 60.01 ± 5.67 | 52.24 ± 2.54 |
| H2GCN | 84.81 ± 6.94 | 86.64 ± 4.63 | 35.83 ± 0.96 | 37.95 ± 1.89 | 58.27 ± 2.63 | 82.08 ± 4.71 | 53.10 ± 1.23 |
| APPNP | 81.93 ± 5.77 | 85.48 ± 4.58 | 35.90 ± 0.96 | 39.08 ± 1.76 | 57.80 ± 2.47 | 81.92 ± 6.12 | 53.06 ± 1.90 |
| GPR-GNN | 79.44 ± 5.17 | 84.46 ± 6.36 | 35.11 ± 0.82 | 32.33 ± 2.42 | 46.76 ± 2.10 | 79.91 ± 6.60 | 52.74 ± 1.88 |
| GAT | 55.21 ± 5.70 | 52.80 ± 6.11 | 29.04 ± 0.66 | 40.00 ± 0.99 | 59.32 ± 1.54 | 61.89 ± 6.08 | 51.28 ± 1.79 |
| +CSA-I | 56.17 ± 5.32 | 53.23 ± 6.28 | 29.03 ± 0.79 | 40.51 ± 0.98 | 60.73 ± 1.35 | 62.75 ± 6.32 | 51.67 ± 1.62 |
| +CSA-II | **58.21 ± 4.79** | **54.35 ± 6.54** | 29.71 ± 0.74 | 41.02 ± 1.23 | **61.31 ± 1.13** | **64.26 ± 5.21** | **52.20 ± 1.74** |
| +CSA-III | 58.04 ± 5.27 | 53.98 ± 6.30 | **29.72 ± 0.86** | **41.38 ± 1.19** | 61.20 ± 1.37 | 63.58 ± 6.03 | 52.13 ± 1.83 |
| FAGCN | 82.54 ± 6.89 | 82.84 ± 7.95 | 34.85 ± 1.24 | 42.55 ± 0.86 | 61.21 ± 3.13 | 79.24 ± 9.92 | 54.35 ± 1.11 |
| +CSA-I | 82.65 ± 7.11 | 83.37 ± 7.79 | 34.77 ± 0.95 | 42.55 ± 0.74 | 61.86 ± 2.98 | 80.01 ± 9.72 | 54.44 ± 1.18 |
| +CSA-II | 83.29 ± 6.80 | 83.11 ± 8.26 | 34.88 ± 0.86 | 42.58 ± 0.93 | 61.74 ± 3.39 | **81.35 ± 9.68** | 54.45 ± 1.23 |
| +CSA-III | **84.72 ± 6.71** | **84.23 ± 7.21** | **35.12 ± 0.98** | **43.38 ± 1.02** | **62.52 ± 3.20** | 80.94 ± 9.77 | **55.16 ± 0.97** |
| WRGAT | 83.62 ± 5.50 | 86.98 ± 3.78 | 36.53 ± 0.77 | 48.85 ± 0.78 | 65.24 ± 0.87 | 81.62 ± 3.90 | 54.76 ± 1.12 |
| +CSA-I | 83.69 ± 5.63 | 87.23 ± 3.94 | 36.55 ± 0.93 | **49.46 ± 0.74** | 65.36 ± 1.05 | 81.88 ± 3.93 | 54.86 ± 1.31 |
| +CSA-II | 83.76 ± 5.61 | 87.02 ± 3.55 | 36.47 ± 0.74 | 48.93 ± 0.89 | 65.33 ± 0.92 | **82.76 ± 3.67** | 54.97 ± 1.23 |
| +CSA-III | **84.88 ± 5.23** | **87.86 ± 3.77** | **36.89 ± 0.72** | 49.43 ± 0.88 | **66.02 ± 1.01** | 82.43 ± 4.00 | **55.33 ± 1.18** |

Table 1: Classification accuracy on heterophily datasets. CSA-I, CSA-II, and CSA-III indicate our three counterfactual schemes respectively.

the behavior of MLP does not seem to be a wise choice for these GAT variants to improve their attentions. Inspired by the self-boosting concept [Pi *et al.*, 2016] widely used in machine learning, we leverage the historical attention map as the counterfactual to urge the factual one keep refining itself. The specific formulation is written as follows:

$$\hat{a} \sim A_{hist}, \tag{7}$$

where $A_{hist}$ denotes the historical attention map (e.g., the attention map from the last update iteration).

## 5 Experiment

In this section, we conduct extensive node classification experiments to evaluate the performance of CSA. Specifically, we 1) validate the effectiveness of CSA on three popular GAT variants using a wide range of datasets, including both homophily and heterophily scenarios; 2) compare CSA with other attention improvement baselines of GAT to show the superiority of our method; 3) show that CSA can induce better attention that improve the robustness of GAT; 4) test CSA's sensitivity to hyper-parameters; 5) analyze the influences of CSA in feature space; and 6) examine the performances of some special cases of CSA.

### 5.1 Datasets

For heterophily scenario, we select seven standard benchmark datasets: Wisconsin, Cornell, Texas, Actor, Squirrel, Chameleon, and Crocodile. Wisconsin, Cornell, and Texas collected by Carnegie Mellon University are published in WebKB[1]. Actor [Pei *et al.*, 2020] is the actor-only subgraph sampling from a film-director-actor-writer network. Squirrel, Chameleon, and Crocodile are datasets collected from the English Wikipedia [Rozemberczki *et al.*, 2021]. We summarize the basic attributions for each dataset in Table 1. $\mathcal{H}(\mathcal{G})$ is the homophily ratio [Zhu *et al.*, 2020], where $\mathcal{H}(\mathcal{G}) \to 1$ represents extreme homophily and vice versa.

For homophily scenario, two large datasets released by Open Graph Benchmark (OGB)[2] [Hu *et al.*, 2020]: ogbn-products and ogbn-arxiv, are included in our experiments, together with two small-scale homophily graph datasets: Cora and Citeseer [McCallum *et al.*, 2000]. Similarly, the attribution of the dataset is summarized in Table 2.

### 5.2 Experimental Setup

We employ popular node classification models in our experiments as the baselines: GCN [Kipf and Welling, 2016], GAT [Veličković *et al.*, 2017], SGCN [Wu *et al.*, 2019], FAGCN [Bo *et al.*, 2021], GPR-GNN [Chien *et al.*, 2020], H2GCN [Zhu *et al.*, 2020], WRGAT [Suresh *et al.*, 2021], APPNP [Gasteiger *et al.*, 2018] and UniMP [Shi *et al.*, 2020]. We also present the performance of MLPs, serving as a strong non-graph-based baseline. Due to page limit, we only select four models: GAT, FAGCN, WRGAT and UniMP to examine the effectiveness of CSA. These models ranges from the classic ones to the latest ones, and are considered as representatives for state-of-the-art node classification models. One thing to be noted here is that for all these models, we implement CSA only in their first layers to avoid excessive computational cost.

In our experiments, each GNN is run with the best hyperparameters if provided. We set the same random seed for each

[1]http://www.cs.cmu.edu/ webkb/

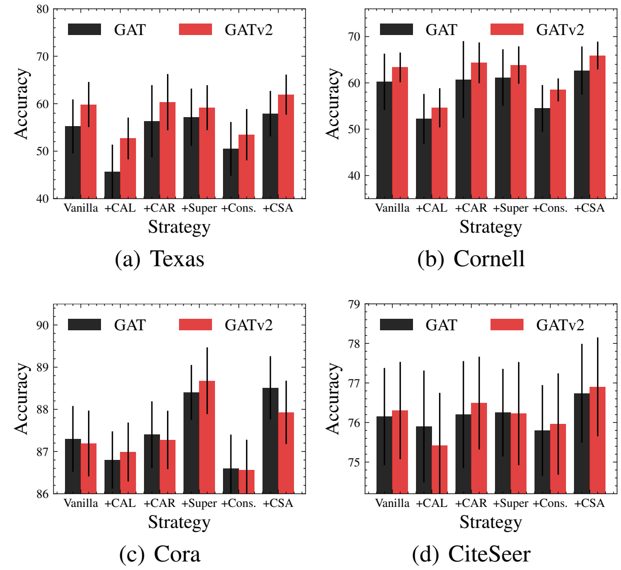
[2]https://ogb.stanford.edu/

| Models | Cora | CiteSeer | ogbn-products | ogbn-arxiv |
|---|---|---|---|---|
| $\mathcal{H}(\mathcal{G})$ | 0.81 | 0.74 | 0.81 | 0.66 |
| **#Nodes** | 2,708 | 3,327 | 2,449,029 | 169,343 |
| **#Edges** | 5,278 | 4,467 | 61,859,140 | 1,166,243 |
| **#Classes** | 7 | 7 | 47 | 40 |
| **#Features** | 1433 | 3703 | 100 | 128 |
| GAT | 86.21 ± 0.78 | 75.73 ± 1.23 | 77.02 ± 0.63 | 70.96 ± 0.14 |
| +CSA-I | 86.16 ± 0.95 | 76.81 ± 1.29 | 77.28 ± 0.69 | 71.08 ± 0.14 |
| +CSA-II | 86.89 ± 0.64 | 76.53 ± 1.18 | 77.44 ± 0.63 | 71.05 ± 0.14 |
| +CSA-III | **87.86 ± 0.87** | **77.72 ± 1.25** | **78.36 ± 0.72** | **71.20 ± 0.16** |
| UniMP | 86.89 ± 0.90 | 75.14 ± 0.68 | 81.37 ± 0.47 | 72.92 ± 0.10 |
| +CSA-I | 87.47 ± 0.87 | 75.89 ± 0.73 | 81.55 ± 0.62 | 72.94 ± 0.10 |
| +CSA-II | 85.62 ± 0.73 | 75.87 ± 0.72 | 81.39 ± 0.47 | 72.96 ± 0.10 |
| +CSA-III | **88.64 ± 1.28** | **77.61 ± 0.82** | **82.24 ± 0.63** | **73.08 ± 0.11** |

Table 2: Classification accuracy on homophily datasets.

| Models | Texas | Cornell | ogbn-arxiv |
|---|---|---|---|
| GAT | 55.21 ± 5.70 | 61.89 ± 6.08 | 70.96 ± 0.14 |
| +CSA (Last) | 57.83 ± 4.65 | 63.52 ± 5.34 | 71.08 ± 0.15 |
| +CSA (Pure) | 55.79 ± 5.05 | 61.97 ± 5.18 | 70.96 ± 0.14 |
| +CSA (Ours) | **58.21 ± 4.79** | **64.26 ± 5.21** | **71.20 ± 0.16** |

Table 3: Comparison with the heuristic causal strategies.




(a) Texas (b) Cornell

(c) Cora (d) CiteSeer

Figure 4: Comparison with different GATs promotion strategies.

model and dataset for reproducibility. The reported results are in the form of mean and standard deviation, calculated from 10 random node splits (the ratio of train/validation/test is 48%/32%/20% from [Pei *et al.*, 2020]). Our experiments are conducted on a GPU server with eight NVIDIA DGX A100 graphics cards, and the codes are implemented using Cuda Toolkit 11.5, PyTorch 1.8.1 and torch-geometric 2.0.1.

### 5.3 Performance Analysis

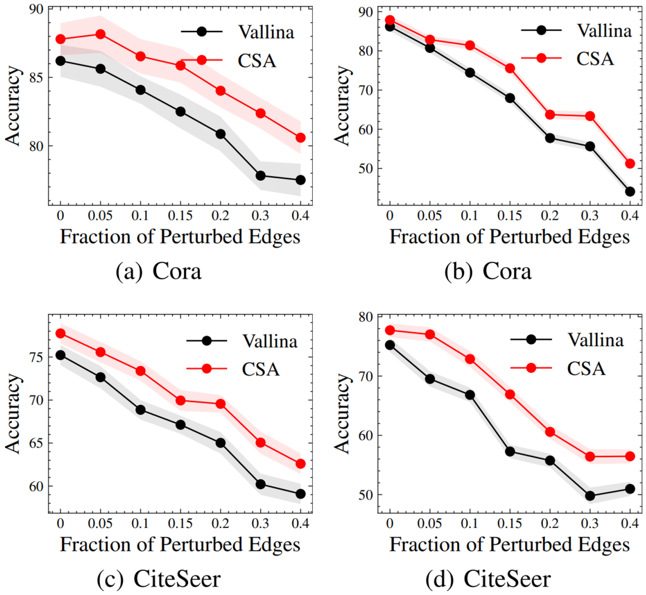
Table 1 and Table 2 provide the test accuracy of different GNNs in different variants of CSA in the supervised node classification task. A graph's homophily level is the average of its nodes' homophily levels. CSA achieves the best in terms of the vanilla one across all datasets. In particular, the highest improved datasets are Texas, Wisconsin, and Cornell. By observing the performance of MLPs, we can see that the common ground of those three datasets contains distinguishable features and a large part of non-homophilous edges. In the meanwhile, the performance of CSA is proportional to the modeling ability. The mechanism behind CSA is to extend the causal effect between nodes representation and final prediction. Therefore, CSA owns limited performance when the node's representations are chaotic. Our experiments highlight that I) The model, which is already better than MLP, does not improve much in CSA-II, while CSA-III improves it relatively more. This is because in those datasets, the graph structure can provide meaningful information, so that CSA-III have more advantages. II) The dataset, which has distinctive features indicated by the performance of MLPs, is more satisfied CSA-II. Similarly, in this scene, the features can be more informative. III) The random strategy (CSA-I) relatively inferior to others, since the distribution is hard to control and tend to generate worst attention map, whereby the regularization is vanished.

### 5.4 Comparison with Attention Promotion Baselines

We here apply multiple attention promotion baselines: CAL [Sui *et al.*, 2022], CAR [Wu *et al.*, 2022a], Super [Kim and Oh, 2022], Constraint interventions [Wang *et al.*, 2019] and the result is shown in Figure 4. Among them, CAL is a method for graph property prediction that relies on an alternative formulation of causal attention with interventions on implicit representations. We adapted CAL for node classification by removing its final pooling layer. Super is well-known as SuperGAT, a method seeking to constrain node feature vectors through a semi-supervised edge prediction task. CAR aligns the attention mechanism with the causal effects of active interventions on graph connectivity in a scalable manner. Constraint method has two auxiliary losses: graph structure-based constraint and class boundary constraint. The results on four datasets are shown in Figure 4. While CAL, CAR, and CSA have related goals of enhancing graph attention using concepts from causal theory, CAL uses abstract perturbations on graph representation to perform causal interventions, and CAR employs an edge intervention strategy that enables causal effects to be computed scalable, while our method does not exert any assumptions and constraints on GATs, compared with CAL and CAR. Therefore, CSA tends to own good generalization ability. In terms of SuperGAT and Constraint method, since there is a trade-off between node classification and regularization. For example, SuperGAT implies that it is hard to learn the relational importance of edges by simply optimizing graph attention for link prediction.

### 5.5 CSA Provides Robustness

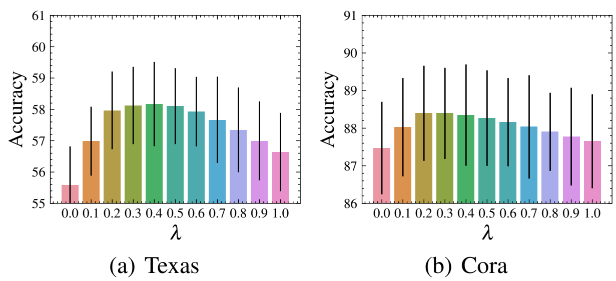
In this section, we systematically study the performance of CSA on robustness against the input perturbations including feature and edge perturbations. Following [Stadler *et al.*, 2021], we conduct node-level feature perturbations by replac-




(a) Cora (b) Cora

(c) CiteSeer (d) CiteSeer

Figure 5: The robust performance on the node and edge perturbations.



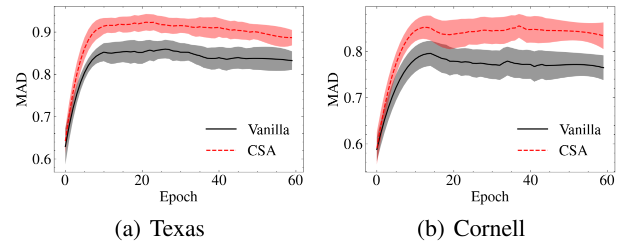

(a) Texas (b) Cora

Figure 6: Hyper-parameter analysis on GAT.

ing them with the noise sampled from the Bernoulli distribution with $p = 0.5$ and edge perturbations by stochastically generating the edges. According to the performance shown in Figure 5, CSA produces robust performance on input perturbations. Figure 5 demonstrate that CSA in higher noise situations achieves more robust results than in lower noise scenes on both node and edge perturbations with perturbation percentages ranging from $0\%$ to $40\%$.

### 5.6 Hyper-parameter Analysis

We analyze the sensitivity of $\lambda$ and plot node classification performance in Figure 6. For $\lambda$, there is a specific range that maximizes test performance in all datasets. The performance in Texas is the highest when $\lambda$ is $0.4$, but the difference is relatively small compared to Cora. We observe that there is an optimal level of causal supervision for each dataset, and using too large $\lambda$ degrades node classification performance. Since Cora owns friendly neighbors, its performance is less sensitive than Texas. Based on this, we can also see that Texas relatively needs a larger regularization.




(a) Texas (b) Cornell

Figure 7: Mean Average Distance among node representations of Last GAT layer.

### 5.7 Pairwise Distance among Classes

To further evaluate whether the good performance of CSA can be contributed to the mitigation of lacking supervision, we visualize the pairwise distance of the node representations among classes learned by CSA and vanilla GAT. Following [Stadler *et al.*, 2021], we calculate the Mean Average Distance (MAD) with cosine distance among node representations for the last layer. The larger the MAD is, the better the node representations are. Results are reported in Figure 7. It can be observed that the node representations learned by CSA keep having a large distance throughout the optimization process, indicating relief of lacking supervision issues. On the contrary, GAT suffers from severely indistinguishable representations of nodes.

### 5.8 Comparison with Heuristic Causal Strategies.

To validate the effectiveness of CSA, we compare it with two heuristic causal designs (Last and Pure) that 1) directly estimate the total causal effect by subtracting between the model's and the counterfactual's output in the final layer; 2) replace the attention with the static aggregate weights (i.e., each node allocates the same weight). The results are shown in Table 3. We observe that their performance outperforms vanilla one, but is still inferior to ours. In terms of Last, the major difference is whether to explicitly estimate causal effect or not. In our framework, we plug the MLPs into the hidden layers to precisely estimate the causal effect for each layer. Regarding Pure, our strategy can provide more strong baselines, leading to better regularization.

## 6 Conclusion

We introduced CSA, a counterfactual-based regularization scheme that can be applied to graph attention architectures. Unlike other causal approaches, we first built the causal graph of GATs in a general way and do not impose any assumptions and constraints on GATs. Subsequently, we introduce an efficient scheme to directly estimate the causal effect of attention in hidden layers. Applying it to both homophilic and heterophilic node-classification tasks, we found accuracy improvements and robustness in almost all circumstances. We performed three variants of counterfactual attention strategies and found that they can adapt to different situations, respectively.

## Contribution Statement

Hongjun Wang and Jiyuan Chen contributed equally to the work. Lun Du and Xuan Song are the corresponding authors. This work is done during Hongjun Wang's internship at Microsoft Research Asia.

## References


[Battaglia *et al.*, 2016] Peter Battaglia, Razvan Pascanu, Matthew Lai, Danilo Jimenez Rezende, et al. Interaction networks for learning about objects, relations and physics. *Advances in neural information processing systems*, 29, 2016.

[Bo *et al.*, 2021] Deyu Bo, Xiao Wang, Chuan Shi, and Huawei Shen. Beyond low-frequency information in graph convolutional networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 3950–3957, 2021.

[Brody *et al.*, 2021] Shaked Brody, Uri Alon, and Eran Yahav. How attentive are graph attention networks? *arXiv preprint arXiv:2105.14491*, 2021.

[Chien *et al.*, 2020] Eli Chien, Jianhao Peng, Pan Li, and Olgica Milenkovic. Adaptive universal generalized pagerank graph neural network. *arXiv preprint arXiv:2006.07988*, 2020.

[Cinelli *et al.*, 2019] Carlos Cinelli, Daniel Kumor, Bryant Chen, Judea Pearl, and Elias Bareinboim. Sensitivity analysis of linear structural causal models. In *International conference on machine learning*, pages 1252–1261. PMLR, 2019.

[Defferrard *et al.*, 2016] Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. *Advances in neural information processing systems*, 29, 2016.

[Du *et al.*, 2022] Lun Du, Xiaozhou Shi, Qiang Fu, Xiaojun Ma, Hengyu Liu, Shi Han, and Dongmei Zhang. Gbk-gnn: Gated bi-kernel graph neural networks for modeling both homophily and heterophily. In *Proceedings of the ACM Web Conference 2022*, pages 1550–1558, 2022.

[Feng *et al.*, 2021] Fuli Feng, Weiran Huang, Xiangnan He, Xin Xin, Qifan Wang, and Tat-Seng Chua. Should graph convolution trust neighbors? a simple causal inference method. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1208–1218, 2021.

[Gao and Ji, 2019] Hongyang Gao and Shuiwang Ji. Graph representation learning via hard and channel-wise attention networks. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 741–749, 2019.

[Gasteiger *et al.*, 2018] Johannes Gasteiger, Aleksandar Bojchevski, and Stephan Günnemann. Predict then propagate: Graph neural networks meet personalized pagerank. *arXiv preprint arXiv:1810.05997*, 2018.

[Gilmer *et al.*, 2017] Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Neural message passing for quantum chemistry. In *International conference on machine learning*, pages 1263–1272. PMLR, 2017.

[Hoshen, 2017] Yedid Hoshen. Vain: Attentional multi-agent predictive modeling. *Advances in Neural Information Processing Systems*, 30, 2017.

[Hou *et al.*, 2022] Yifan Hou, Jian Zhang, James Cheng, Kaili Ma, Richard TB Ma, Hongzhi Chen, and Ming-Chang Yang. Measuring and improving the use of graph information in graph neural networks. *arXiv preprint arXiv:2206.13170*, 2022.

[Hu *et al.*, 2020] 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.

[Huang *et al.*, 2023] Jincheng Huang, Lun Du, Xu Chen, Qiang Fu, Shi Han, and Dongmei Zhang. Robust mid-pass filtering graph convolutional networks. In *Proceedings of the ACM Web Conference 2023*, pages 328–338, 2023.

[Kim and Oh, 2020] Dongkwan Kim and Alice Oh. How to find your friendly neighborhood: Graph attention design with self-supervision. In *ICLR*, 2020.

[Kim and Oh, 2022] Dongkwan Kim and Alice Oh. How to find your friendly neighborhood: Graph attention design with self-supervision. *arXiv preprint arXiv:2204.04879*, 2022.

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

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

[Kumar *et al.*, 2010] M Pawan Kumar, Benjamin Packer, and Daphne Koller. Self-paced learning for latent variable models. In *NIPS*, volume 1, page 2, 2010.

[Lee *et al.*, 2019] John Boaz Lee, Ryan A Rossi, Sungchul Kim, Nesreen K Ahmed, and Eunyee Koh. Attention models in graphs: A survey. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 13(6):1–25, 2019.

[Li *et al.*, 2022] Xiang Li, Renyu Zhu, Yao Cheng, Caihua Shan, Siqiang Luo, Dongsheng Li, and Weining Qian. Finding global homophily in graph neural networks when meeting heterophily. *arXiv preprint arXiv:2205.07308*, 2022.

[Liu *et al.*, 2021] Meng Liu, Zhengyang Wang, and Shuiwang Ji. Non-local graph neural networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

[McCallum *et al.*, 2000] Andrew Kachites McCallum, Kamal Nigam, Jason Rennie, and Kristie Seymore. Automating the construction of internet portals with machine learning. *Information Retrieval*, 3:127–163, 2000.

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

[Pearl, 2014] Judea Pearl. Interpretation and identification of causal mediation. *Psychological methods*, 19(4):459, 2014.

[Pearl, 2022] Judea Pearl. Direct and indirect effects. In *Probabilistic and Causal Inference: The Works of Judea Pearl*, pages 373–392. 2022.

[Pei *et al.*, 2020] Hongbin Pei, Bingzhe Wei, Kevin Chen-Chuan Chang, Yu Lei, and Bo Yang. Geom-gcn: Geometric graph convolutional networks. *arXiv preprint arXiv:2002.05287*, 2020.

[Pi *et al.*, 2016] Te Pi, Xi Li, Zhongfei Zhang, Deyu Meng, Fei Wu, Jun Xiao, Yueting Zhuang, et al. Self-paced boost learning for classification. In *IJCAI*, pages 1932–1938, 2016.

[Ribeiro *et al.*, 2016] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144, 2016.

[Rozemberczki *et al.*, 2021] Benedek Rozemberczki, Carl Allen, and Rik Sarkar. Multi-scale attributed node embedding. *Journal of Complex Networks*, 9(2):cnab014, 2021.

[Santoro *et al.*, 2017] Adam Santoro, David Raposo, David G Barrett, Mateusz Malinowski, Razvan Pascanu, Peter Battaglia, and Timothy Lillicrap. A simple neural network module for relational reasoning. *Advances in neural information processing systems*, 30, 2017.

[Shi *et al.*, 2020] Yunsheng Shi, Zhengjie Huang, Shikun Feng, Hui Zhong, Wenjin Wang, and Yu Sun. Masked label prediction: Unified message passing model for semi-supervised classification. *arXiv preprint arXiv:2009.03509*, 2020.

[Stadler *et al.*, 2021] Maximilian Stadler, Bertrand Charpentier, Simon Geisler, Daniel Zügner, and Stephan Günnemann. Graph posterior network: Bayesian predictive uncertainty for node classification. *Advances in Neural Information Processing Systems*, 34:18033–18048, 2021.

[Stolberg *et al.*, 2004] Harald O Stolberg, Geoffrey Norman, and Isabelle Trop. Randomized controlled trials. *AJR Am J Roentgenol*, 183(6):1539–44, 2004.

[Sui *et al.*, 2022] 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*, pages 1696–1705, 2022.

[Sun *et al.*, 2021] Chuxiong Sun, Hongming Gu, and Jie Hu. Scalable and adaptive graph neural networks with self-label-enhanced training. *arXiv preprint arXiv:2104.09376*, 2021.

[Suresh *et al.*, 2021] Susheel Suresh, Vinith Budde, Jennifer Neville, Pan Li, and Jianzhu Ma. Breaking the limit of graph neural networks by improving the assortativity of graphs with local mixing patterns. *arXiv preprint arXiv:2106.06586*, 2021.

[Thekumparampil *et al.*, 2018] Kiran K Thekumparampil, Chong Wang, Sewoong Oh, and Li-Jia Li. Attention-based graph neural network for semi-supervised learning. *arXiv preprint arXiv:1803.03735*, 2018.

[VanderWeele, 2016] Tyler J VanderWeele. Explanation in causal inference: developments in mediation and interaction. *International journal of epidemiology*, 45(6):1904–1908, 2016.

[Vaswani *et al.*, 2017] 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.

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

[Wang *et al.*, 2019] Guangtao Wang, Rex Ying, Jing Huang, and Jure Leskovec. Improving graph attention networks with large margin-based constraints. *arXiv preprint arXiv:1910.11945*, 2019.

[Wang *et al.*, 2021] Yujing Wang, Yaming Yang, Jiangang Bai, Mingliang Zhang, Jing Bai, Jing Yu, Ce Zhang, Gao Huang, and Yunhai Tong. Evolving attention with residual convolutions. In *International Conference on Machine Learning*, pages 10971–10980. PMLR, 2021.

[Wang *et al.*, 2022a] Hongjun Wang, Jiyuan Chen, Zipei Fan, Zhiwen Zhang, Zekun Cai, and Xuan Song. St-expertnet: A deep expert framework for traffic prediction. *IEEE Transactions on Knowledge and Data Engineering*, 2022.

[Wang *et al.*, 2022b] Hongjun Wang, Jiyuan Chen, Tong Pan, Zipei Fan, Boyuan Zhang, Renhe Jiang, Lingyu Zhang, Yi Xie, Zhongyi Wang, and Xuan Song. Easy begun is half done: Spatial-temporal graph modeling with st-curriculum dropout. *arXiv preprint arXiv:2211.15182*, 2022.

[Wu *et al.*, 2019] Felix Wu, Amauri Souza, Tianyi Zhang, Christopher Fifty, Tao Yu, and Kilian Weinberger. Simplifying graph convolutional networks. In *International conference on machine learning*, pages 6861–6871. PMLR, 2019.

[Wu *et al.*, 2022a] Alexander P Wu, Thomas Markovich, Bonnie Berger, Nils Hammerla, and Rohit Singh. Causally-guided regularization of graph attention improves generalizability. *arXiv preprint arXiv:2210.10946*, 2022.

[Wu *et al.*, 2022b] Shiwen Wu, Fei Sun, Wentao Zhang, Xu Xie, and Bin Cui. Graph neural networks in recommender systems: a survey. *ACM Computing Surveys*, 55(5):1–37, 2022.

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

[Ye and Ji, 2021] Yang Ye and Shihao Ji. Sparse graph attention networks. *IEEE Transactions on Knowledge and Data Engineering*, 2021.

[Ying *et al.*, 2021] Chengxuan Ying, Tianle Cai, Shengjie Luo, Shuxin Zheng, Guolin Ke, Di He, Yanming Shen, and Tie-Yan Liu. Do transformers really perform badly for graph representation? *Advances in Neural Information Processing Systems*, 34:28877–28888, 2021.

[You *et al.*, 2020] Yuning You, Tianlong Chen, Zhangyang Wang, and Yang Shen. When does self-supervision help graph convolutional networks? In *international conference on machine learning*, pages 10871–10880. PMLR, 2020.

[Zhang and Chen, 2018] Muhan Zhang and Yixin Chen. Link prediction based on graph neural networks. In *NeurIPS*, pages 5165–5175, 2018.

[Zhang *et al.*, 2018] Jiani Zhang, Xingjian Shi, Junyuan Xie, Hao Ma, Irwin King, and Dit-Yan Yeung. Gaan: Gated attention networks for learning on large and spatiotemporal graphs. *arXiv preprint arXiv:1803.07294*, 2018.

[Zhang *et al.*, 2020] Kai Zhang, Yaokang Zhu, Jun Wang, and Jie Zhang. Adaptive structural fingerprints for graph attention networks. In *International Conference on Learning Representations*, 2020.

[Zhang *et al.*, 2022] Wentao Zhang, Ziqi Yin, Zeang Sheng, Yang Li, Wen Ouyang, Xiaosen Li, Yangyu Tao, Zhi Yang, and Bin Cui. Graph attention multi-layer perceptron. *arXiv preprint arXiv:2206.04355*, 2022.

[Zhao *et al.*, 2022] Tong Zhao, Gang Liu, Daheng Wang, Wenhao Yu, and Meng Jiang. Learning from counterfactual links for link prediction. In *International Conference on Machine Learning*, pages 26911–26926. PMLR, 2022.

[Zhu *et al.*, 2020] Jiong Zhu, Yujun Yan, Lingxiao Zhao, Mark Heimann, Leman Akoglu, and Danai Koutra. Beyond homophily in graph neural networks: Current limitations and effective designs. *Advances in Neural Information Processing Systems*, 33:7793–7804, 2020.