计算机科学
COMPUTER SCIENCE

基于因果推断的图注意力网络

张涛，程毅飞，孙欣煦

引用本文

张涛，程毅飞，孙欣煦.基于因果推断的图注意力网络[J]．计算机科学, 2023, 50(6A): 220600230-9.

ZHANG Tao, CHENG Yifei, SUN Xinxu. Graph Attention Networks Based on Causal Inference[J]. Computer Science, 2023, 50(6A): 220600230-9.

计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.220600230

基于因果推断的图注意力网络

张 涛 程毅飞 孙欣煦
燕山大学信息科学与工程学院 河北 秦皇岛 066004
河北省信息传输与信号处理重点实验室 河北 秦皇岛 066004

摘 要 图注意力网络(Graph Attention Network,GAT)是一种重要的图神经网络,在分类任务中有着广泛的应用。但是当图中邻域节点受到干扰时,模型分类准确度会受影响而降低。对此,提出一种基于因果推断的图注意力网络(Causal Graph Attention Network,C-GAT)以提升网络的鲁棒性。该模型首先计算目标节点的邻域与其标签之间的因果权重,并以此对邻域进行因果采样;然后计算采样后邻域与目标节点之间的注意力系数;最后根据注意力系数对邻域信息进行加权聚合,获得目标节点的嵌入特征。在Cora,Pubmed和Citeseer数据集上的实验结果表明,在无扰动的情况下,C-GAT的分类性能与经典模型持平;在有扰动的情况下,相比于经典模型,C-GAT的分类准确度至少提升了6%,在鲁棒性和时间成本上有着较好的平衡。

关键词:图注意力网络;因果推断;注意力机制;因果权重
中图法分类号 TP183

Graph Attention Networks Based on Causal Inference

ZHANG Tao,CHENG Yifei and SUN Xinxu
School of Information Science and Engineering,Yanshan University,Qinhuangdao,Hebei 066004,China
Hebei Key Laboratory of Information Transmission and Signal Processing,Qinhuangdao,Hebei 066004,China

Abstract Graph attention network(GAT) is an important graph neural network with a wide range of applications in classification tasks. However,when the neighborhood nodes in the graph are disturbed,the model classification accuracy will be affected and degraded. In response,a graph attention network based on causal inference named causal graph attention network(C-GAT) is proposed to improve the robustness of the network. The model first calculates the causal weights between the neighborhood of the target node and its label and uses them to sample the neighborhood. Then the attention coefficient between the sampled neighborhood and the target node is calculated. Finally,the embedding features of the target nodes are obtained by weighted aggregation of the neighborhood information based on the attention coefficients. Experimental results on the Cora,Pubmed and Citeseer datasets show that the classification performance of C-GAT is on par with the classical model in the case of no perturbation. In the presence of perturbations,the classification accuracy of C-GAT improves by at least 6% compared to the classical model,with a better balance of robustness and time cost.

Keywords Graph attention networks,Causal inference,Attention mechanism,Causal weight

1 引言

图注意力网络由Velikovi等[1]于2018年提出,通过将注意力机制引入到图神经网络模型中,解决了图卷积神经网络难以为图中每个邻居节点分配不同权重的问题[2],在行人识别[3]、声纹识别[4]、情感分析[5]、文字识别[6]、高光谱图像[7]以及场景图生成[8]等领域获得了广泛的应用。

与此同时,针对GAT性能的改进也层出不穷。针对不同特征情况,Yang等[9]采用计算中心节点与其远邻居之间不同长度的最短路径的方法,在一层网络内计算高阶的节点特征和注意力系数,从而提高了分类性能。Zhou等[10]提出了SAN(Structural Attention Network),将图中节点的输出特征表示为多阶特征串联,从而可以区分多阶结构。Jing等[11]提出了DeepGCN(Deep Graph Convolutional Networks),用于提取深层抽象特征并探索恒指数据间的内部关系。由于使用了新的相似性度量方法,稀疏矩阵的存储虽然使用了所有光谱信息,但占用了大量空间且计算的复杂性仍然很高。针对图结构的不同特点,Gu等[12]提出了GANR(Graph Attention Based Network Representation),利用图注意力架构将图结构作为监督学习信息,不仅能够学习高质量的节点表示,还可以提取注意力权重并将其应用于节点分类任务,但是调整超参数的过程十分繁琐。Zhang等[13]提出了一种编码复杂的拓扑和结构信息的自适应结构指纹模型,充分利用节点周围的结构化细节信息,缓解了GAT中固有的过光滑问题。在

基金项目:国家自然科学基金(62176229);河北省自然科学基金(F2020203010)
This work was supported by the National Natural Science Foundation of China(62176229) and Natural Science Foundation of Hebei Province,China(F2020203010).
通信作者:张涛(zhtao@ysu.edu.cn)

改进注意力机制方面，Yang 等[14]提出了 Graph-CAT(Graph Co-Attention Networks)，基于两种不同但互补的注意力方案执行局部和全局属性增强，弥补了 GAT 应用于局部属性增强时的局限性。Xie 等[15]提出了 M-GAT(Multi-view Graph Attention Networks)，采用了一种新的注意力机制，考虑每个视图的不同属性，由各视图之间的信息处理节点，可以更高效地编码多视图网络。考虑到时间问题，Zhang 等[16]提出了 AGC-Seq2Seq(Attention Graph Convolutional Sequence-to-Sequence Model)，在所提出的深度学习框架中将空间和时间通过 Seq2Seq 模型建模，进一步捕捉交通模式的时间异质性。Ji 等[17]提出了一种 hop-aware 注意力监督方法，模拟退火学习策略，沿训练时间线平衡节点分类和跳跃感知注意系数这两个学习任务，提高模型的分类性能，在标记节点较少的图归纳任务中比较有效。在扩展图注意力计算方法上，Zhang 等[18]提出一种用于图学习的多核集成注意力方法，将自动多核学习纳入图网络的工作中。Song 等[19]提出了两个端到端的可训练算子——超图卷积和超图注意力，但是将这两个运算符与其他图神经网络相结合仍是一个挑战。此外，在提升性能的同时，网络的鲁棒性也开始受到重视。就图数据而言，对图结构信息的有效利用以及对噪声数据扰动的鲁棒性也十分重要[20]。Feng 等[21]提出了 GRAND(Graph Random Neural Networks)，通过使用随机传播策略删除无效节点，有效地提高了图神经网络的鲁棒性和抗平滑性，但其时间成本相对较高。

在机器学习领域，因果推断被证明是提高系统鲁棒性的有效方法[22]。Judea 为解决图形因果模型中的因果推理问题，开发了 do 算子[23]等有效的工具。为了进一步扩展因果推理在机器学习中的应用，Bernhard 等[24]将其与机器学习的迁移、泛化联系了起来。在医学领域，为了解决基于关联关系的医学诊断可能会导致错误的诊断结果的问题，Richens 等[25]从因果推断的角度提出一种反事实诊断算法，有效提高了诊断分类的鲁棒性。Little 等[26]提出根据变量之间的因果关系对数据进行重新采样，使任何机器学习算法都能够做出因果稳健的预测。Liu 等[27]改进了基于原子动作的贝叶斯模型，解决了网络结构固定的问题，能更精准地识别复杂活动。Zhang 等[28]提出了 CAT(Causality by Attribute Topology)，从对象和属性之间关系的角度分析了中医临床数据中各症候间的因果关系，获得了与典籍相似的演化结论。

受以上研究启发，本文提出了基于因果推断的图注意力网络 C-GAT。首先，将因果推断引入到 GAT 中，分析了其中存在的因果混淆问题。然后，构建了一个特殊的有向因果图，提出了一种计算目标节点邻域之间因果权重的算法，并通过该权重引导对邻域信息的因果采样。最后，根据采样结果对邻域进行加权聚合，得到目标节点嵌入表示。为验证所提模型的鲁棒性，本文在 Cora，Pubmed 和 Citeseer 数据集上开展了实验，实验结果表明 C-GAT 的鲁棒性相比于传统模型有显著的优势，同时在时间成本上有着良好的平衡。

2 GAT 中的因果混淆问题

在 GAT 中，用 $G=(V,E)$ 来表示一个图，其中 V 表示节点集合，E 表示边集合。N 表示集合 V 中节点的个数，任取集合 V 中两个节点 v_i 和 v_j，$e_{ij}=(v_i,v_j)\in E$ 表示在 v_i 和 v_j 之间存在一条边 e_{ij}。$x_r\in R^D$ 表示集合 V 中节点 v_r 的特征，节点特征矩阵表示为 $X\in R^{N\times D}$，其中 D 是节点特征向量的维数。Y 表示所有节点标签的集合，节点 v_r 的标签表示为 y_r，节点 v_r 的邻域定义为：$N(v)=\{u\in V|(v_r,u)\in E\}$。

GAT 由若干个功能相同的注意力层构成，通过在注意力层上计算注意力系数来量化节点之间的重要性，然后使用注意力系数对邻域特征加权聚合，获得节点的特征嵌入表示[1]。然而，对于任意 $v_i,v_j\in N(v_r)$，总会存在一条无向路径$\langle v_i,v_r,v_j\rangle$，即 $N(v_r)\cup v_r$ 中的节点可能会相互影响，如图 1 所示。如果在加权聚合过程中选取了 $N(v_r)$中的所有节点，当其中任意一个节点被扰动所干扰时，该扰动不仅会影响该节点本身，还会传播到 $N(v_r)\cup v_r$ 中其他节点，这就导致了 $N(v_r)\cup v_r$ 内节点之间的重要性被破坏，注意力系数 α 发生改变。由于分类结果取决于对 $N(v_r)\cup v_r$ 中节点信息与注意力系数 α 的加权聚合，因此扰动也会影响最后的分类结果，降低模型的分类鲁棒性。

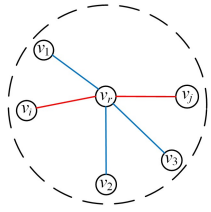


图 1 邻域中存在的无向路径

Fig. 1 Undirected paths existing in the neighborhood

3 基于因果推断的图注意力网络模型

为了解决意外扰动带来的因果问题，本文提出在 GAT 中使用因果推断来减少扰动的影响。理论上，在 GAT 中一个注意力层可以有多个注意力头，后文中主要使用第一个注意力头，其他注意力头可由此推导得出。图 2 给出了因果注意力头为 1 的 C-GAT 获得节点嵌入表示的过程，图中红色点代表目标节点，橙色点代表邻域节点，α 为注意力系数，W 为权重矩阵，$\vec{a}$ 为权重向量。整个训练过程分为 3 个步骤：

(1)对输入数据进行因果采样；

(2)对因果采样后的因果性邻域进行注意力系数的计算；

(3)根据注意力系数对邻域进行加权聚合。

3.1 因果结构建立

首先需要在 GAT 的注意力层上建立一个特殊的有向因果图，并在此基础上研究因果后门准则的适用性。选取节点 v_r 的邻域 $N(v_r)$，并将节点 v_r 的特征 x_r 作为注意力层的输入。在注意力层进行加权聚合的过程中，节点与其标签之间的因果信息并没有被考虑进去。而从节点分类角度来看 GAT 时，任意一个节点 $v_i\in N(v_r)$都是用来去预测其标签 y_i 的。这就意味着标签 y_i 作为结果，而节点 v_i 作为其原因，即存在一条有向的路径从 v_i 指向 y_i。同理，任意一个节点 v_i 和$\bar{V}_i(v_r)=v_r+N(v_r)\backslash v_i$ 中的节点都是标签集合 Y 中其标签的原因。所以这里存在着两条因果路径：$v_i\rightarrow Y$，$\bar{V}_i(v_r)\rightarrow Y$(如图 3 所示)和 $v_i\rightarrow y_i$(如图 4(a)所示)。

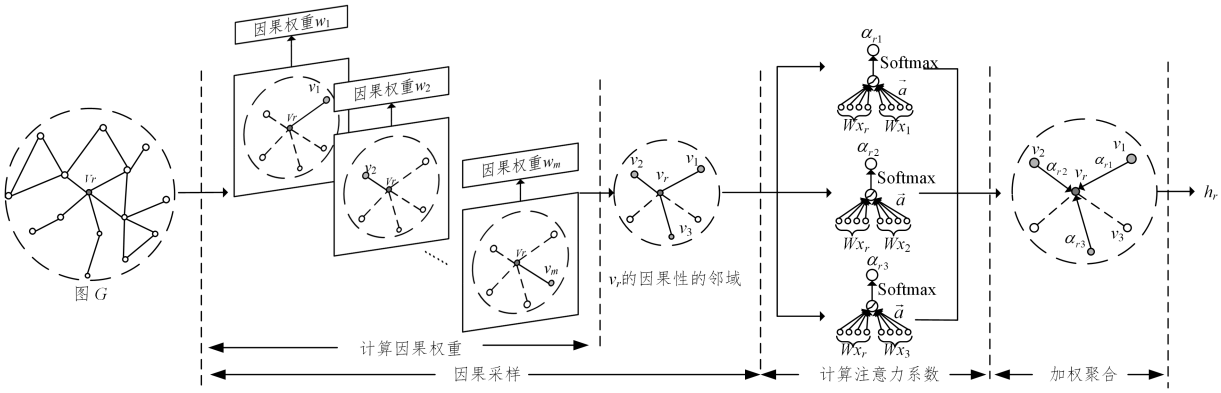


图 2　训练过程中注意力头为 1 时 C-GAT 获得节点特征嵌入表示的流程

Fig. 2　Process of C-GAT obtains the node feature embedding representation when attention head is 1 during the training process

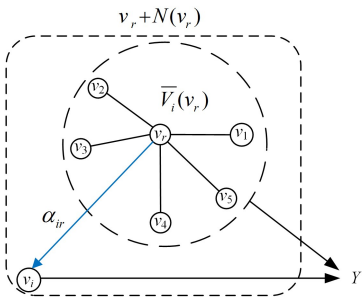


图 3　节点与标签集合 Y 的关系

Fig. 3　Relationship between nodes and label sets Y

与此同时，着眼于整个图 G 时，任何一个节点都是其邻域内其他节点的原因和结果。因为对于任何一个节点 $v_j \in \bar{V}_i(v_r)$，总会存在一条路径$\langle v_i, v_r, v_j \rangle$。但是从注意力层对邻域节点加权聚合的角度来看，若选取节点 v_i 作为注意力层的输入，v_i 只会被其邻居节点影响而不会影响其邻居节点，这意味着$\bar{V}_i(v_r)$中的任何干扰都会通过 GAT 注意力层中的加权聚合而传播到节点 v_i 上。因此$\bar{V}_i(v_r)$便可作为节点 v_i 的原因，即节点 v_i 是被$\bar{V}_i(v_r)$所影响的结果，故在注意力层加权聚合的过程中存在一条因果路径：$\bar{V}_i(v_r) \rightarrow v_i$（如图 4(a)所示）。

在图 3 中存在 $v_i \rightarrow y_i$ 和$\bar{V}_i(v_r) \rightarrow v_i$ 两条因果路径的前提下，由于 y_i 作为 v_i 的结果，那么在$\bar{V}_i(v_r)$中的任何对 v_i 的影响都会干扰到 v_i 对标签 y_i 的预测。也就是说，可以将$\bar{V}_i(v_r)$视作 v_i 的标签 y_i 的间接原因，即存在因果路径：$\bar{V}_i(v_r) \rightarrow y_i$，如图 4(a)所示。

至此，可以将一个无向不包含因果信息的图形数据转化为一个包含因果信息的有向因果图，建立出一个特殊的因果结构模型。

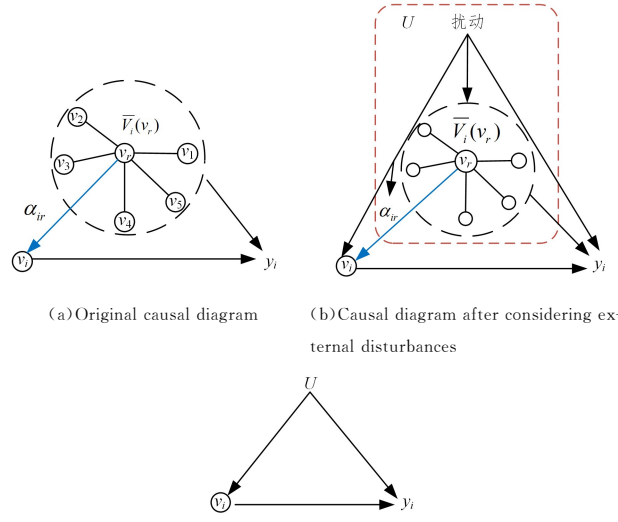
3.2　基于因果权重的采样

考虑外界扰动时，图 4(a)中的因果图可以转化为图 4(b)。一般情况下，对图数据的扰动可能直接或间接地导致图中节点的损坏，注意力系数被破坏并且预测结果被干扰。因此，可将外界的扰动视为 v_i，$\bar{V}_i(v_r)$和 y_i 的一部分原因。由于$\bar{V}_i(v_r)$会被扰动所影响，并且此影响会传播到 v_i 与 y_i 上，因此可使用 U 来表示扰动和被扰动所干扰的$\bar{V}_i(v_r)$的混合体。那么，图 4(b)就可以转化为如图 4(c)所示的最简因果模型。从 v_i 到 y_i 的介入分布公式可以表示为：

$$p(y_i \mid do(v_i)) = \int p(y_i \mid pa(y_i)) \prod_{j \in \varepsilon} p(j \mid pa(j)) \mathrm{d}\varepsilon \tag{1}$$

其中，$pa(y_i)$和 $pa(v_i)$分别表示因果图中 y_i 和 v_i 的父集合，$\varepsilon = pa(y_i) \backslash v_i$ 作为被边缘化的意外变量。对于图 4(c)中的因果图，式(1)可以写为：

$$p(y_i \mid do(v_i)) = \int p(y_i \mid v_i, U) p(U) \mathrm{d}U \tag{2}$$



(a) Original causal diagram　　(b) Causal diagram after considering external disturbances

(c) Abstract causaldiagrabased on(b)

图 4　基于 C-GAT 的因果图

Fig. 4　C-GAT based causal diagram

由于此时在图 4(c)中存在着 3 条因果路径将 y_i，U 和 v_i 联系在一起，并且它们之间满足：

(1)把 v_i 视作图中的一个被动变量时，U 中没有 v_i 的子类；

(2)U 阻断了 $v_i \rightarrow y_i$ 间的所有因果路径。可见，路径 $v_i \leftarrow U \rightarrow y_i$ 之间的关系满足后门准则[22]。在此基础上，可将式(2)近似为：

$$p(y_i \mid do(v_i)) \approx \frac{1}{N} \sum_{n=1}^{N} K[y_i - y_n] \frac{K[v_i - v_n]}{\hat{p}(v_i \mid U)} \tag{3}$$

其中，$K[\cdot]$是由联合和边际再生核希尔伯特空间(RKHS)函数得到的核密度估计(KDE)。接下来令：

$$w_i = \frac{1}{N \hat{p}(v_i \mid U)} \tag{4}$$

可将式(3)转化为：

$$p(y_i|do(v_i)) \approx w_i \sum_{n=1}^{N} K[y_i - y_n] \cdot K[v_i - v_n] \tag{5}$$

其中，$K[\cdot]$为连续变量下的狄拉克 δ 函数，离散变量下的离散克罗内克 δ 函数。因此式(5)中等号右边的部分表示了节点和标签的加权采样规则。式(5)可以进一步表示为：

$$p(y_i|do(v_i)) = \begin{cases} w_i, & y_i = y_n, v_i = v_n \\ 0, & \text{其他情况} \end{cases} \tag{6}$$

式(1)－式(6)描述了从 v_i 到 y_i 的介入分布近似到 w_i 的过程。w_i 称为因果权重，而通过因果权重来进行采样的方式称为因果采样。

3.3 计算节点特征嵌入

当一个节点 v_r 被输入到图注意力层后，首先对其进行邻域因果抽样，即对于目标节点的所有邻居，获得从其自身到其标签的干预分布的因果权重，然后根据因果权重进行采样。

当完成对目标节点邻域的因果采样之后，对因果采样后的邻域进行注意力权重的计算，然后通过注意力权重对因果采样后的邻域信息加权聚合获得目标节点的嵌入表示。令因果采样后的邻域集合为 C，此时在 C-GAT 中 v_i 的特征嵌入可以表示为：

$$h_{v_i} = \sigma\left(\sum_{j \in C} \alpha_{ij} W x_j\right) \tag{7}$$

为了稳定自我注意力机制的学习过程，C-GAT 采用了多注意力头的方式。若一个注意力层有 K 个注意力头，意味着 K 个独立注意机制执行式(7)的变换，最后将嵌入特征串联起来，得到最终的特征嵌入表示：

$$h_{v_i} = \Big\Vert_{k=1}^{K} \sigma\left(\sum_{j \in C} \alpha_{ij}^{k} W^k x_j\right) \tag{8}$$

其中，$\Vert$ 代表拼接操作，α_{ij}^{k} 为第 k 个注意头计算的归一化注意力系数，W^k 是相应的输入线性变换的权重矩阵。多注意力头的 C-GAT 获得节点嵌入表示的过程如算法 1 所示。

算法 1 多注意力的 C-GAT 的节点嵌入获得

输入：图 G=(V,E)，节点 v∈V，节点集合 V 内节点的数量 N，n∈{1,…,N}，扰动 U，因果采样个数 s∈Z^+

输出：节点的嵌入表示 h_v

1. for n=1,…,N do
2. 获得节点 v_n 的邻域：$N(v_n)$.
3. for $v_i \in N(v_r)$ do
4. 计算 v_i 和 U 的联合概率密度分布下的 KDEs：$\hat{p}(v_i, U) \leftarrow \frac{1}{N}\sum_{n=1}^{N} K[v_i - v_n] K[U - U_n]$.
5. 计算 U 的概率密度分布的 KDEs：$\hat{p}(U) \leftarrow \frac{1}{N}\sum_{n=1}^{N} K[U - U_n]$.
6. U 的条件概率分布：$\hat{p}(v_i|U) \leftarrow \frac{\hat{p}(v_i, U)}{\hat{p}(U)}$.
7. 获得因果采样权重 w_i：$w_i \leftarrow \frac{1}{N\,\hat{p}(v_i|U)}$.
8. end
9. 根据因果权重 w_i 在 $N(v_n)$ 中采样节点：$C=\{v_1, \cdots, v_s\}$.
10. for $v_j \in C$, do
11. 计算 v_n 与 v_j 之间的注意力系数 α_{nj}：

$$\alpha_{nj} \leftarrow \frac{\exp(\text{LeakyReLU}(\vec{a}^{T}[Wx_n \Vert Wx_j]))}{\sum_{k \in C} \exp(\text{LeakyReLU}(\vec{a}^{T}[Wx_n \Vert Wx_k]))}$$

12. 获得节点 v_n 的特征嵌入表示 h_{v_n}：$h_{v_n} \leftarrow \sum_{k=1}^{K} \sigma\left(\sum_{j \in C} \alpha_{nj}^{k} W^k x_j\right)$
13. end

最后，在参数学习阶段，C-GAT 作为一种通用的图表示学习模型，在分类任务中可以使用交叉熵损失函数。在前向传播时，使用 Softmax 函数来对图注意力层的输出 h_v 进行归一化，并将归一化后的向量 z_i 作为前向传播的输出。向量 z_i 可以表示为：

$$z_i = \text{Softmax}(h_{v_i}) = \frac{\exp(h_{v_i})}{\sum_{c=1}^{M} \exp(h_{v_i c})} \tag{9}$$

其中，M 表示所有节点的类别的个数，$h_{v_i c}$ 代表节点嵌入表示向量 h_{v_i} 中第 c 个元素。

在反向传播阶段，使用 Adam 优化器[29]来优化权重参数矩阵 W 和权重向量 $\vec{a}$。在分类任务中的损失函数可以表示为：

$$L = -\sum_{i=1}^{N}\sum_{j=1}^{M} y_{ij} \log(z_{ij}) \tag{10}$$

其中，y_i 是节点 v_i 的真实标签，y_{ij} 是指示变量，当标签 $y_i = j$ 时 $y_{ic} = 1$，z_{ij} 表示节点 v_i 属于类别 j 的概率。

4 实验结果与分析

4.1 实验数据集与基线设置

实验采用了 3 个应用广泛的引文网络数据集(Cora，Pubmed，Citeseer)作为基准数据集，数据集的统计信息如表 1 所列。

表 1 实验中所使用的数据集概要

Table 1 Summary of datasets used in the experiments

数据集	Cora	Pubmed	Citeseer
节点数量	2708	19717	3327
边的数量	5429	44338	4732
类别数量	7	3	6
特征向量的维度	1433	500	3703

基线选择经典的图神经网络(包括 GCN[30]，GraphSAGE[31]，GAT[1])，此外还选择了 GraphHop[32]；同时为了更好地对比 C-GAT 分类鲁棒性，选取了最新的具有良好鲁棒性的 GRAND 作为基线。各模型实验超参数依照原文献设置如下。

C-GAT：网络层数＝2，dropout 概率＝0.6，第一层的注意力头数＝8，在 Cora 和 Citeseer 数据集上，输出注意力头数＝1，L2 正则化参数＝5×10^{-4}；在 Pubmed 数据集上，输出注意力头＝8，L2 正则化参数＝1×10^{-3}。

GAT：网络层数＝2，dropout 概率＝0.6，注意力头＝8，各注意力头计算特征数＝8，在 Cora 和 Citeseer 数据集上，输出注意力头＝1，L2 正则化参数＝5×10^{-4}；在 Pubmed 数据集上，输出注意力头＝8，L2 正则化参数＝1×10^{-3}。

GRAND：dropnode 概率＝0.5，隐藏层大小＝32，L2 正则化参数＝5×10^{-4}；在 Cora 数据集上，传播步长大小＝8，一致性正则化系数＝1；在 Pubmed 数据集上，传播步长大小＝2，一致性正则化系数＝0.7；在 Citeseer 数据集上，传播步长大小＝5，一致性正则化系数＝1。

GraphHop：L2 正则化参数＝5×10^{-5}，在 Cora 和 Pubmed 数据集上，$T=0.1$，$\alpha=10$，$\beta=1$；Citeseer 数据集上，$T=0.1$，$\alpha=1$，$\beta=1$。

GCN：dropout 概率＝0.5，L2 正则化参数＝5×10^{-4}，隐藏层大小＝16。

GraphSAGE：网络层数＝2，邻域采样个数：第一层＝25，第二层＝10。

为了验证模型分类鲁棒性，根据之前的研究[21]，引入扰动来评估模型鲁棒性。生成一个与节点特征矩阵 $X\in R^{N\times D}$ 大小相同的伯努利矩阵 X_B。被扰动干扰的节点特征矩阵 F_B 可以表示为：

$$F_B=X\oplus X_B \tag{11}$$

其中，$\oplus$代表异或运算，定义扰动比率为：

$$1-\frac{1}{N\times D}\sum_{i=1}^{N}\sum_{j=1}^{D}X_B(i,j) \tag{12}$$

从每个类选取 20 个节点作为训练集，并在 1 000 个测试节点上评估分类性能。此外，训练数据中的任何节点都有关于其特征和标签的信息。在模型训练过程中，使用 Adam 优化更新神经网络的权值，最大迭代次数为 20 次。评价指标选取常用的准确度(accuracy)和 F1-score，其定义为：

$$accuracy=\frac{TP+TN}{TP+TN+FP+FN} \tag{13}$$

$$F1\text{-}score=2\cdot\frac{precision\cdot recall}{precision+recall} \tag{14}$$

其中，TP 是预测正确的分类正确节点数，TN 是预测为错误的分类错误节点数，FP 是预测为正确的分类错误节点数，FN 是预测为错误的分类正确节点数，$precision=\frac{TP}{TP+FP}$，$recall=\frac{TP}{TP+FN}$。

本次实验的硬件平台是：英特尔 CPU(i7-9700@3 GHz)，16 GB 内存(RAM)，操作系统为 Windows10。

4.2　无扰动情况下的实验

对每种算法进行 10 次重复实验并取平均值。表 2 列出了 C-GAT 与基线模型在准确度和 F1-Score 上的性能对比，其中 GraphSAGE-Pool 表示聚合函数为池化函数的 GraphSAGE。

表 2　无扰动情况下各模型在 3 个数据集上的实验结果

Table 2　Experimental results of each model on three data sets without perturbation

模型		Cora		Pubmed		Citeseer	
		accuracy	*F1-score*	*accuracy*	*F1-score*	*accuracy*	*F1-score*
对比基线方法	GraphSAGE-Pool	0.796	0.785	0.781	0.782	0.709	0.657
	GraphSAGE-Mean	0.792	0.781	0.785	0.780	0.700	0.670
	GraphSAGE-LSTM	0.794	0.785	0.779	0.776	0.696	0.647
	GCN	0.806	0.797	**0.787**	0.782	0.708	0.703
	GraphHop	0.802	0.782	0.771	0.779	0.703	0.672
	GRAND	0.809	**0.804**	0.786	0.783	0.713	0.696
对比 GAT	GAT	0.811	0.801	0.785	**0.785**	0.714	**0.719**
	C-GAT▲	**0.813**	0.803	0.784	0.782	**0.716**	0.708

注：▲表示本文方法

从表 2 中可以看出，与原始 GAT 模型相比，C-GAT 在 Cora 和 Citeseer 数据集上的性能表现略优于 GAT，在 Pubmed 数据集上的表现与 GAT 基本持平，这表明在无扰动的情况下，相比于只关注节点之间重要性的注意力机制，加入因果推断进行因果采样具有一定优势。对比其他基线，C-GAT 在 3 个数据集上的表现基本优于 GraphHop，GCN 以及 3 个不同聚合函数的 GraphSAGE。此外，C-GAT 在 Cora 和 Pubmed 数据集上的表现与 GRAND 持平，在 Citeseer 数据集上略优于 GRAND。

4.3　有扰动情况下的实验

为了评估鲁棒性，本节中添加扰动，并增加扰动的比率与基线方法进行对比。表 3－表 5 分别列出了 C-GAT 与基线方法在 3 个实验数据集上，不同扰动比率下的分类准确度。

表 3　Cora 数据集上，有扰动情况下各个模型的分类准确度

Table 3　Classification accuracy of each model with disturbances on Cora dataset

测试数据	模型	扰动比率				
		0.1	0.2	0.3	0.4	0.5
测试数据添加扰动	C-GAT▲	**0.729**	0.708	**0.689**	**0.671**	**0.655**
	GAT	0.717	0.679	0.647	0.606	0.591
	GRAND	0.726	**0.712**	0.685	0.670	0.650
	GraphHOP	0.679	0.662	0.648	0.635	0.625
	GCN	0.714	0.672	0.634	0.598	0.576
	GraphSAGE-Pool	0.694	0.673	0.626	0.593	0.588
	GraphSAGE-Mean	0.68	0.663	0.628	0.590	0.587
	GraphSAGE-LSTM	0.689	0.662	0.625	0.592	0.589
测试数据不添加扰动	C-GAT▲	**0.798**	0.781	**0.767**	**0.753**	**0.742**
	GAT	0.780	0.753	0.732	0.721	0.719
	GRAND	0.791	**0.782**	0.764	0.751	0.741
	GraphHOP	0.755	0.741	0.729	0.718	0.706
	GCN	0.754	0.742	0.736	0.728	0.720
	GraphSAGE-Pool	0.753	0.738	0.727	0.716	0.712
	GraphSAGE-Mean	0.754	0.735	0.724	0.713	0.705
	GraphSAGE-LSTM	0.758	0.741	0.732	0.726	0.714

注：▲表示本文方法

表 4 Pubmed 数据集上，有扰动情况下各个模型的分类准确度

Table 4 Classification accuracy of each model with disturbances on Pubmed dataset

测试数据	模型	扰动比率				
		0.1	0.2	0.3	0.4	0.5
测试数据添加扰动	C-GAT▲	**0.738**	**0.715**	**0.687**	**0.668**	**0.649**
	GAT	0.704	0.673	0.636	0.607	0.581
	GRAND	0.737	0.709	0.684	0.662	0.641
	GraphHOP	0.683	0.654	0.623	0.592	0.563
	GCN	0.702	0.670	0.632	0.601	0.578
	GraphSAGE-Pool	0.710	0.667	0.628	0.599	0.575
	GraphSAGE-Mean	0.690	0.662	0.630	0.591	0.578
	GraphSAGE-LSTM	0.691	0.654	0.632	0.586	0.569
测试数据不添加扰动	C-GAT▲	**0.768**	**0.756**	**0.748**	**0.741**	**0.735**
	GAT	0.751	0.743	0.734	0.728	0.717
	GRAND	0.764	0.752	0.746	0.739	0.730
	GraphHOP	0.753	0.741	0.730	0.719	0.708
	GCN	0.750	0.742	0.732	0.724	0.712
	GraphSAGE-Pool	0.749	0.744	0.731	0.723	0.715
	GraphSAGE-Mean	0.752	0.733	0.726	0.715	0.708
	GraphSAGE-LSTM	0.747	0.736	0.728	0.714	0.709

注：▲表示本文方法

表 5 Citeseer 数据集上，有扰动情况下各个模型的分类准确度

Table 5 Classification accuracy of each model with disturbances on Citeseer dataset

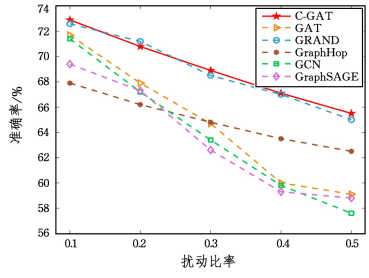
测试数据	模型	扰动比率				
		0.1	0.2	0.3	0.4	0.5
测试数据添加扰动	C-GAT▲	**0.642**	**0.624**	**0.601**	**0.589**	**0.562**
	GAT	0.629	0.610	0.571	0.529	0.509
	GRAND	0.638	0.622	0.597	0.584	0.560
	GraphHOP	0.620	0.581	0.556	0.539	0.515
	GCN	0.622	0.598	0.561	0.514	0.494
	GraphSAGE-Pool	0.624	0.604	0.563	0.516	0.497
	GraphSAGE-Mean	0.617	0.581	0.56	0.512	0.495
	GraphSAGE-LSTM	0.618	0.589	0.557	0.516	0.492
测试数据不添加扰动	C-GAT▲	0.697	0.681	**0.666**	**0.645**	**0.636**
	GAT	0.673	0.646	0.635	0.621	0.603
	GRAND	**0.698**	**0.682**	0.664	0.644	0.634
	GraphHOP	0.688	0.662	0.641	0.625	0.606
	GCN	0.662	0.632	0.622	0.612	0.594
	GraphSAGE-Pool	0.666	0.631	0.620	0.611	0.591
	GraphSAGE-Mean	0.668	0.639	0.619	0.607	0.585
	GraphSAGE-LSTM	0.661	0.635	0.615	0.606	0.590

注：▲表示本文方法

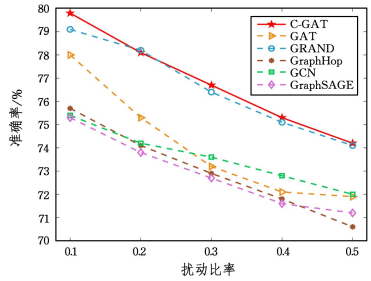
从表 3－表 5 可以看出，当扰动比率逐渐增加的时候，C-GAT 在 Cora 和 Pubmed 数据集上的分类性能明显优于 GraphSAGE，GCN，GAT 以及 GraphHop，并且与 GRAND 相比分类性能基本持平，下降趋势较缓和，表现出了优秀的鲁棒性。当训练数据和测试数据中都添加扰动时，C-GAT 的分类准确度高于基线模型，在 Cora，Pubmed 和 Citeseer 数据集上的分类准确度最大提升分别为 6.5%，6.8%和 6.0%；当只对训练数据添加扰动时，C-GAT 的分类准确度最大提升分别为 3.5%，1.8%和 3.5%。这些提升，证明了 C-GAT 因果采样的方式可以有效提高模型在有扰动情况下的鲁棒性。

为了更好地体现 C-GAT 的分类鲁棒性，图 5 给出了在 Cora 数据集上，测试数据添加扰动和测试数据没扰动情况下，随着扰动比率的增加，C-GAT 和其他基线的准确度。另外从表 3－表 5 中可以看出，GraphSAGE-Pool 在 3 个数据集上的表现略优于其他两种聚合函数，所以在下文中用 GraphSAGE-Pool 作为 GraphSAGE 方法的代表进行性能比较。

从图 5 中可以看出，当扰动比率逐渐增加的时候，相较于基线模型，C-GAT 的分类准确度下降趋势较为缓慢。



(a) Test data with perturbation



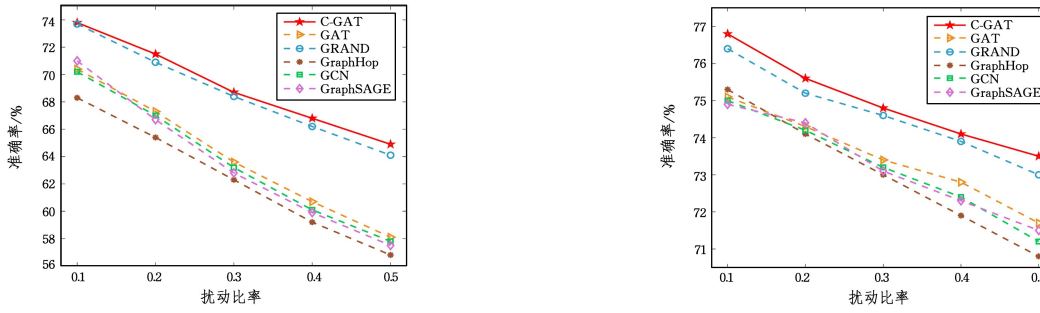
(b) Test data without perturbation

图 5 Cora 数据集上 C-GAT 与基线方法的准确率对比

Fig. 5 Accuracy comparison of C-GAT and baseline method on Cora dataset

此外，随着噪声比率的增加，GRAND 的分类准确度基本与 C-GAT 持平，并且在扰动比率为 0.2 时略优于 C-GAT；但随着噪声比率的逐渐增大，GRAND 的表现逐渐略低于 C-GAT。这是因为 GRAND 通过使用随机传播的方式，让每个节点的特征可以部分或全部随机删除，使每个节点都能够对特定的邻域不敏感，从而提高了 GRAND 的分类鲁棒性[21]。由 C-GAT 与 GRAND 的分类准确度变化趋势可以看出，C-GAT 的分类准确度下降更为平滑，这是因为 GRAND 采用随机传播的方式，虽然能使节点的特征可以部分或全部随机删除，但是在随机删除的时候并没有考虑到节点的因果性，因此受干扰程度较高。而 C-GAT 是根据因果权重来进行采样的，采样的节点之间具有较高的因果性，因此受到扰动干扰的程度比 GRAND 低。这也表明了在有干扰的情况下，结合因果采样的方法比随机传播的方法能提升鲁棒性。图 6 和图 7 分别显示了在 Pubmed 和 Citeseer 数据集上 C-GAT 与基线方法的准确度比较，可以清晰地观察到，C-GAT 的分类鲁棒性也明显高于选取的基线模型，并略优于 GRAND。

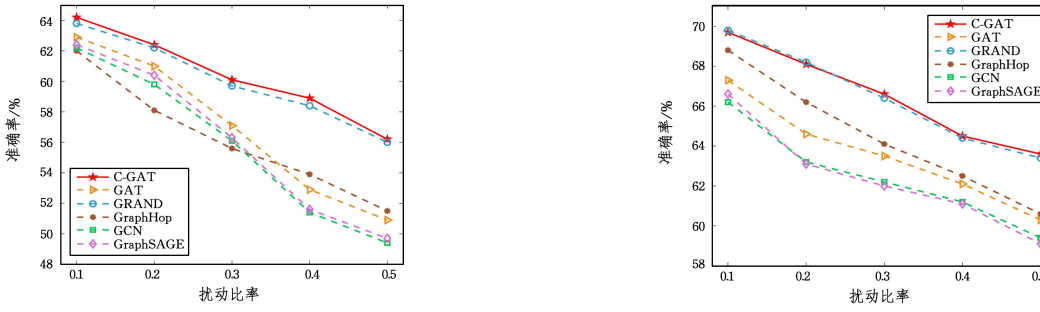
结合表 3－表 5 和图 5－图 7 可以看出，在有扰动干扰时，C-GAT 在 3 个数据集上的分类性能基本全面优于基线方法，并且随着扰动比率的增大，C-GAT 的分类鲁棒性优势越加明显。



(a) Test data with perturbation　　(b) Test data without perturbation

图 6　Pubmed 数据集上 C-GAT 与基线方法的准确率对比

Fig. 6　Accuracy comparison of C-GAT and baseline method on Pubmed dataset



(a) Test data with perturbation　　(b) Test data without perturbation

图 7　Citeseer 数据集上 C-GAT 与基线方法的准确率对比

Fig. 7　Accuracy comparison of C-GAT and baseline method on Citeseer dataset

4.4　模型时间成本分析

本节比较不同模型的实验运行时间，评估 C-GAT 的计算时间成本问题。表 6 列出了 C-GAT 和所选基线模型在训练批次大小为 50 时 1 个阶段下的训练时间。

表 6　C-GAT 和基线模型在 3 个数据集上 1 个阶段下的训练时间

Table 6　Training time of C-GAT and baseline models in one stage on three datasets

（单位：s）

模型	数据集		
	Cora	Pubmed	Citeseer
C-GAT▲	0.461	0.473	0.747
GAT	0.438	0.457	0.700
GRAND	0.617	1.173	0.972
GraphHop	0.323	0.483	0.236
GCN	**0.091**	**0.087**	**0.118**
GraphSAGE-Pool	0.141	0.107	0.184
GraphSAGE-Mean	0.137	0.106	0.174
GraphSAGE-LSTM	0.321	0.241	0.514

注：▲表示本文模型

从表 6 可以看出，在 3 个数据集上，C-GAT 的训练时间比 GAT 分别多出 0.023 s，0.016 s，0.047 s，时间成本增加的比率为 5.2%，3.5%，6.7%。时间成本略有增加的原因是 C-GAT 在计算注意力系数之前要进行因果采样，而因果采样中权重的计算导致了额外的计算复杂度，并且由时间成本增加比率可以看出，时间成本增加不超过 7%，说明了因果采样的计算复杂度低于 GAT 训练阶段的计算复杂度。所有实验模型的时间成本对比如图 8 所示。

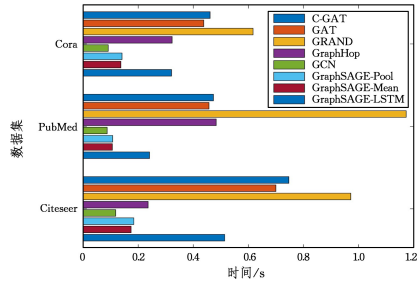


图 8　不同数据集上 C-GAT 和基线方法的时间成本比较

Fig. 8　Time cost comparison of C-GAT and baselines on different datasets

从图 8 可以看出，在 3 个数据集上，C-GAT 的时间消耗明显高于 GCN 与不同聚合函数的 GraphSAGE，这是因为 C-GAT 不仅要在因果采样时计算因果权重，还要计算节点之间的重要性，导致时间成本上升。在所有模型中，GCN 的时间成本最低，因为 GCN 使用了重整化的传播模型，但其鲁棒性明显低于 C-GAT。在上节中，可以看到 GRAND 在有扰动情况下具有较高的性能，但是由于 GRAND 不仅使用随机传播策略来执行图形数据扩充，还要利用一致性正则化来优化不同数据增强中未标记节点的预测一致性，导致时间成本大幅上升，可以看出 GRAND 有着最高的时间成本。在有扰动的情况下，C-GAT 的鲁棒性略优于 GRAND，并且在每个数据集上 C-GAT 的时间成本都小于 GRAND。由此可以得出，C-GAT 实现了在鲁棒性和时间成本之间良好的平衡。

结束语 本文提出了 C-GAT 模型，通过在图注意力层中计算节点与标签之间的因果权重来引导因果采样的方式把因果推断引入到 GAT 模型中，并以此提高模型的分类鲁棒性。通过在 Cora，Pubmed 和 Citeseer 数据集上的实验表明，引入因果推断可以使 C-GAT 比基线方法具有更好的分类鲁棒性，并在性能和时间成本之间实现了良好的平衡。

下一步将针对 C-GAT 当前的局限性开展进一步研究。首先，尽管 C-GAT 对于图数据扰动具有良好的鲁棒性，但是 C-GAT 的运算时间相对较长，如何在保持 C-GAT 的高分类鲁棒性的同时降低计算时间值得研究。其次，在实验过程中，GRAND 由于使用的随机传播策略和一致性正则化的方式体现出了与 C-GAT 不相上下的分类鲁棒性，因此在随机传播策略的过程中加入因果推断可能会进一步提高模型的分类鲁棒性。

参 考 文 献

[1] VELIKOVI P, CUCURULL G, CASANOVA A, et al. Graph Attention Networks[C]// The 6th International Conference on Learning Representations. 2018: 1-12.

[2] TAN Y, WANG J, ZHANG C. Review of Text Classification Methods Based on Graph Convolutional Network[J]. Computer Science, 2022, 49(8): 205-216.

[3] XU S J, LIU Q Y, SHI Y, et al. Person Re-Identification Based on Diversified Local Attention Network[J]. Journal of Electronics & Information Technology, 2022, 44(1): 211-220.

[4] JUNG J, HEO H S, YU H J, et al. Graph Attention Networks for Speaker Verification [C]// IEEE International Conference on Acoustics, Speech and Signal Processing. Ontario: IEEE, 2021: 6149-6153.

[5] HAN H, WU Y H, QIN X Y. An Interactive Graph Attention Networks Model for Aspect-level Sentiment Analysis[J]. Journal of Electronics & Information Technology, 2021, 43(11): 3282-3290.

[6] ZHANG J, LI M X, GAO K S, et al. Word and Graph Attention Networks for Semi-Supervised Classification[J]. Knowledge and Information Systems, 2021, 63: 2841-2859.

[7] CAI W W, WEI Z G. Remote Sensing Image Classification Based on a Cross-Attention Mechanism and Graph Convolution[J]. IEEE Geoscience and Remote Sensing Letters, 2022, 19: 1-5.

[8] ZHOU H, YANG Y Z, LUO T J, et al. A Unified Deep Sparse Graph Attention Network for Scene Graph Generation[J]. Pattern Recognition, 2022, 123: 108367.

[9] YANG Y D, WANG X C, SONG M L, et al. SPAGAN: Shortest Path Graph Attention Network[C]// International Joint Conference on Artificial Intelligence. 2019: 4099-4105.

[10] ZHOU A Z, LI Y F. Structural Attention Network for Graph [J]. Applied Intelligence, 2021, 51(8): 6255-6264.

[11] BAI J, DING B X, XIAO Z, et al, Hyperspectral Image Classification Based on Deep Attention Graph Convolutional Network [J]. IEEE Transactions on Geoscience and Remote Sensing, 2022, 60: 1-16.

[12] GU W W, GAO F, LOU X D, et al. Discovering Latent Node Information by Graph Attention Network[J]. Scientific Reports, 2021, 11(1): 6979.

[13] ZHANG K, ZHU Y, WANG J, et al. Adaptive Structural Fingerprints for Graph Attention Networks[C]// The 8th International Conference on Learning Representations. 2020.

[14] YANG L, LI W X, GUO Y F, et al. Graph-CAT: Graph Co-Attention Networks Via Local and Global Attribute Augmentations[J]. Future Generation Computer Systems, 2021, 118: 170-179.

[15] XIE Y, ZHANG Y Q, GONG M G, et al. Multi-View Graph Attention Networks[J]. Neural Networks, 2020, 132: 180-189.

[16] ZHANG Z C, LI M, LIN X, et al. Multistep Speed Prediction on Traffic Networks: A Deep Learning Approach ConsideringSpatio-Temporal Dependencies[J]. Transportation Research Part C: Emerging Technologies, 2019, 105: 297-322.

[17] JI C J, WANG R X, ZHU R X, et al. Hop-Aware Supervision Graph Attention Networks for Sparsely Labeled Graphs [J/OL]. https://arxiv.org/abs/2004.04333.

[18] ZHANG H M, XU M. Graph Neural Networks with Multiple Kernel Ensemble Attention [J]. Knowledge-Based Systems, 2021, 229: 107299.

[19] BAI S, ZHANG F H, TORR P H S. Hypergraph Convolution and Hypergraph Attention[J]. Pattern Recognition, 2021, 110: 107637.

[20] WANG Z H, SHEN H W, CAO Q, et al. Survey on Graph Classification[J]. Journal of Software, 2022, 33(1): 171-192.

[21] FENG W Z, ZHANG J, DONG Y X, et al. Graph Random Neural Networks for Semi-supervised Learning on Graphs[C]// Advances in Neural Information Processing Systems, 2020, 33: 22092-22103.

[22] JUDEA P. The Seven Tools of Causal Inference with Reflections on Machine Learning[J]. Communications of the ACM, 2019, 62(3): 54-60.

[23] JUDEA P. Causal Inference[M]. Causality: Objectives and Assessment, 2010: 39-58.

[24] SCHOLKOPF B, LOCATELLO F, BAUER S, et al. Toward Causal Representation Learning[J]. Proceedings of the IEEE, 2021, 109(5): 612-634.

[25] RICHENS J G, LEE C M, JOHRI S. Improving the Accuracy of Medical Diagnosis with Causal Machine Learning[J]. Nature Communications, 2020, 11: 47-54.

[26] LITTLE M A, BADAWY R. Causal Bootstrapping [J/OL].

https://arxiv.org/abs/ 1910.09648.

[27] LIU L, WANG S, HU B, et al. Learning structures of interval-based Bayesian networks in probabilistic generative model for human complex activity recognition[J]. Pattern Recognition, 2018, 81: 545-561

[28] ZHANG T, LIU M Q, LIU W Y. The Causality Research Between Syndrome Elements by Attribute Topology[J]. Computational and Mathematical Methods in Medicine, 2018, 2018: 1-12.

[29] KINGMA D P, BA J L. Adam: A Method for Stochastic Optimization[C]// ICIR 2015. 2015.

[30] KIPF T N, WELLING M. Semi-supervised Classification with Graph Convolutional Networks[C]// International Conference on Learning Representations. 2017.

[31] HAMILTON W L, YING R, LESKOVEC J. Inductive Representation Learning on Large Graphs[C]// Proceedings of the 31st International Conference on Neural Information Processing Systems. 2017: 1025-1035.

[32] XIE T, WANG B, JAY K C C. GraphHop: An Enhanced Label Propagation Method for Node Classification[J/OL]. https://arxiv.org/abs/2101.02326v1.

ZHANG Tao, born in 1979, Ph.D, professor, doctoral supervisor, is a member of China Computer Federation. His main research interests include the causal inference, machine learning, and formal concept analysis.