Beijing Forest Studio

北京理工大学信息系统及安全对抗实验中心

基于因果推理的对抗防御方法

硕士研究生 郭汝赞

2025年 11月 09日

问题回溯

- 总结反思

- 相关内容
  - 2024.12.15 吴晓豪：《面向深度学习模型的鲁棒性解释方法研究》
  - 2024.06.03 夏志豪：《图神经网络的反事实解释方法》
  - 2024.01.17 段学明：《 DNN中的理论可解释性》

内容提要

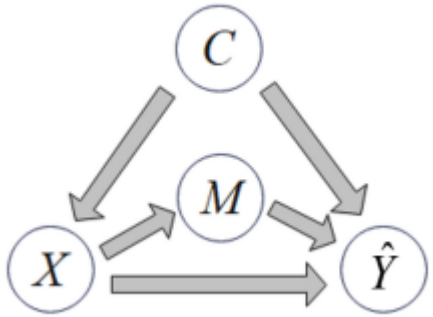
- 预期收获
- 内容引入
- 内涵解析与研究目标
- 研究背景与研究意义
- 研究历史与现状
- 知识基础
- 算法原理
  - CASR
  - CTIoT
- 特点总结与工作展望
- 参考文献

预期收获

- 预期收获
  - 1. 了解基于因果推理的对抗防御的基本概念和问题框架
  - 2. 理解利用因果推理进行对抗防御的必要性
  - 3. 了解因果推理的对抗防御前沿方法

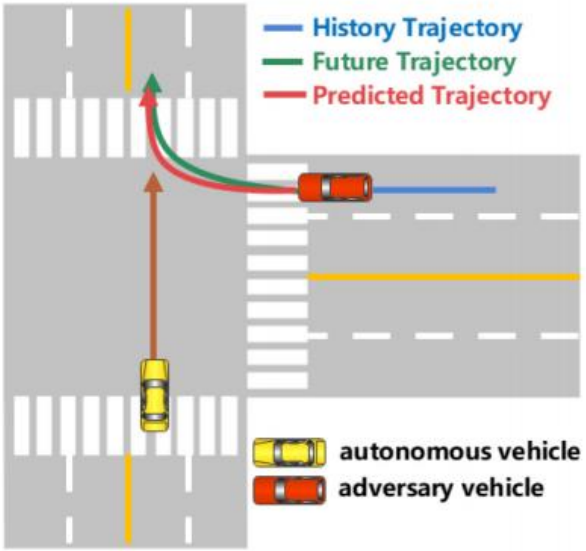
内涵解析与研究目标

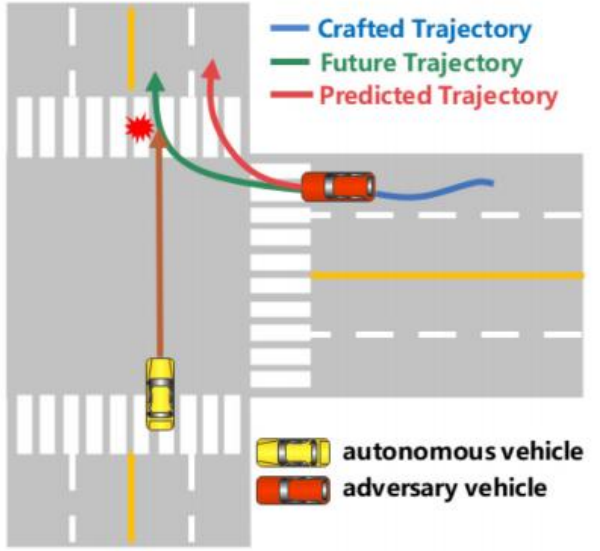
- 研究目标
  - 面向真实部署中的对抗扰动与分布移位，构建在新域、新攻击下仍能稳定工作的模型
  - 结合因果推理与干预学习，识别因果成分与非因果成分，学习不变理据并降低对捷径或伪相关的依赖
- 内涵解析
  - 在不同干预、环境分布下，同时最小化风险期望与方差，获得跨域稳定的外推能力
  - 将可观测输入划分为因果成分与非因果成分，以不变性约束预测，从机制上切断模型对伪相关的依赖，抵抗攻击和变化



研究背景与研究意义

- 研究背景
  - 攻防常态化：AI 系统广泛部署，攻击成本低、影响面广，模型在真实流量下必须面对对抗攻击风险
  - 模型“走捷径”问题：深度模型可能利用伪相关以追求经验风险最小化，表面精度高但对扰动极不稳健





- 研究意义
  - 机制级稳健：通过因果变量分解与外生干预，切断模型对伪相关的依赖，在未知攻击与分布移位下保持稳定性能
  - 跨域与可移植：方法论可迁移到图、时序、文本/多模态与大模型等不同架构与任务，降低复用成本

研究历史

2019 Chao等人利用因果干预理论，通过计算因果效应来量化图像中特定特征对模型预测的真正因果影响；提出了因果效应图，在面对对抗性攻击时表现出更高的敏感性。但该方法评估因果效应需要大量干预操作，且效果依赖于自编码器的重建质量

2020 Zhang等人提出了deep CAMA，该模型显式地建模了数据生成过程中的潜在因果变量，并通过变分推断方法进行训练，使其能够从观测数据中推断出潜在的扰动因素。但该方法未针对类别依赖的扰动或依赖模型梯度的对抗攻击进行专门优化，且获取准确的因果图有一定难度

2021 Chao等人提出了因果推理Q网络，使用一个切换网络结构，根据干扰标签选择不同的Q网络分支进行训练，并在测试阶段通过预测的干扰标签来推断潜在状态，从而保持决策的稳定性。该方法的缺点在于其训练过程中依赖于干扰标签的可用性难以判断

2022 Zhang等人利用作者首次从因果推理的视角系统地分析了深度神经网络的对抗脆弱性问题。作者指出其根源在于模型过度依赖标签与风格变量之间的伪相关，而非真正的因果关系。因此作者提出了CausalAdv方法，通过对齐自然分布和对抗分布来减少这种伪相关的影响。缺点是因果图的构建依赖外部先验知识而非自动学习

2023 Preben等人将四个因果模型统一重构、引入风格解缠度量方法，得到了基于ResNet18的主干架构。实验表明因果信号与混淆信号之间的解缠程度与对抗鲁棒性呈强正相关。但实验仅涵盖了四种模型和三个相对简单的数据集，度量选择和信号定义仍具有一定主观性

2024 Jia等人提出了CASR模型。它利用结构因果模型、干预分布和双级协作优化等方法，显著提高了GNN对数据操纵攻击和OOD偏移的防御能力。但该方案方法复杂度较高、对图结构因果关系做了强假设，泛化性可能较低

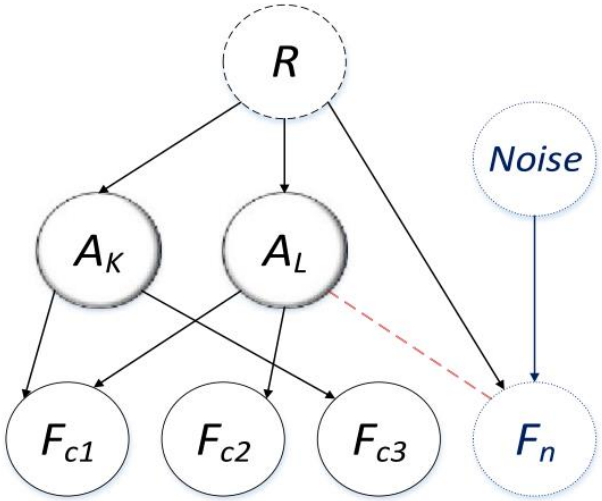
2025 Chao等人通过因果注意力机制学习已知流量的因果特征分布，采用最小最大策略放大正常与异常流量之间的因果特征差异，通过无监督学习对未知攻击进行聚类和增量学习。该方法的缺点在于系统结构较为复杂，计算开销较大

知识基础 因果推理

- 因果推理
  - 不仅仅关注变量之间的相关性，更要揭示因果关系
  - 核心思想：反事实
    - 比较现实世界和反事实世界的差异

$$\text{Causal Effect} = Y(1) - Y(0)$$

$Y(1)$指当干预发生时的结果， $Y(0)$指不干预的结果



    - 对于每个个体，我们只能看到其中一个结果，我们可以使用不同方法来估计反事实结果如随机试验、匹配方法、模型推断等
  - 关键概念：do-干预
    - 被动观察中，变量之间关系会被混杂因素干扰
    - 主动改变某变量，其他的影响因素不变

A Causality-Aligned Structure Rationalization Scheme Against Adversarial Biased Perturbations for Graph Neural Networks

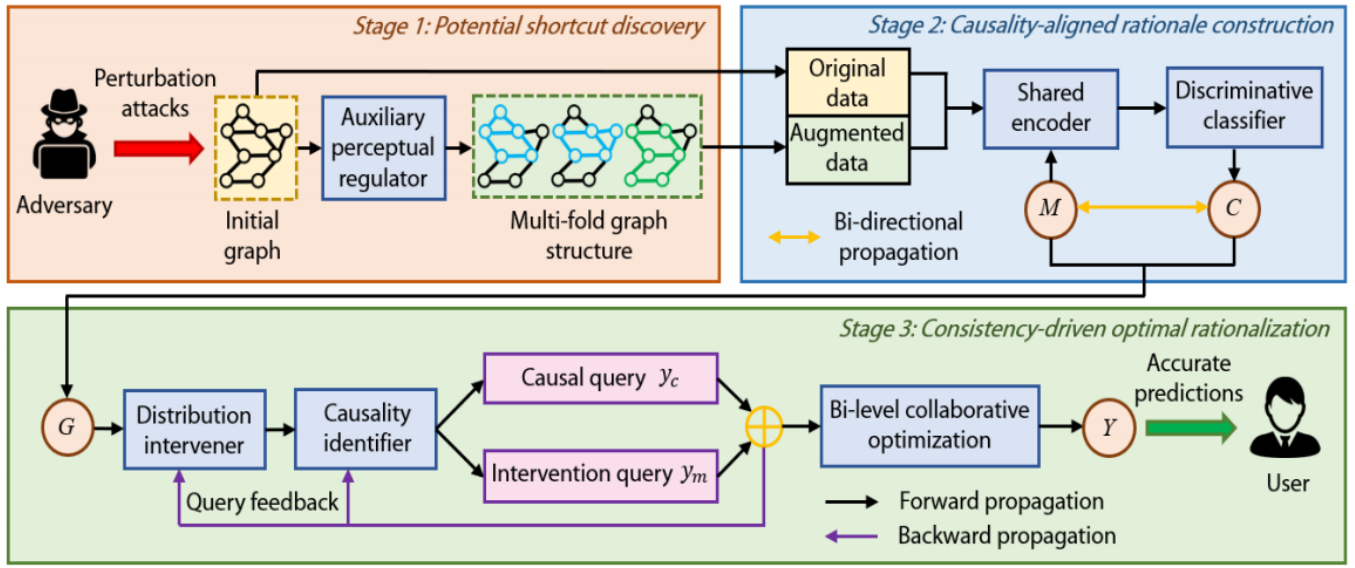
算法原理 TIPO

T	目标	抵抗对GNN的对抗扰动，通过因果对齐结构提升稳健性与泛化能力
I	输入	原始的图数据和初始模型架构
P	处理	1.显式挖掘图中可能存在的特征依赖 2.通过干预分析提取图中稳定的因果子结构 3.使用反馈机制不断优化并强化真正因果结构的稳定性与优先级
O	输出	更强抗干扰的模型

P	问题	现有GNN在面对图结构扰动时常依赖表层关联，难以从复杂邻接结构中抽离出稳定因果要素，导致解释有偏差、泛化能力差、鲁棒性弱
C	条件	图结构的扰动版本与真实标签；能进行因果建模与干预
D	难点	如何联合利用GNN中结构信息与语义表示，构造稳定泛化的因果结构
L	水平	IEEE TIFS 2024 中科院SCI一区

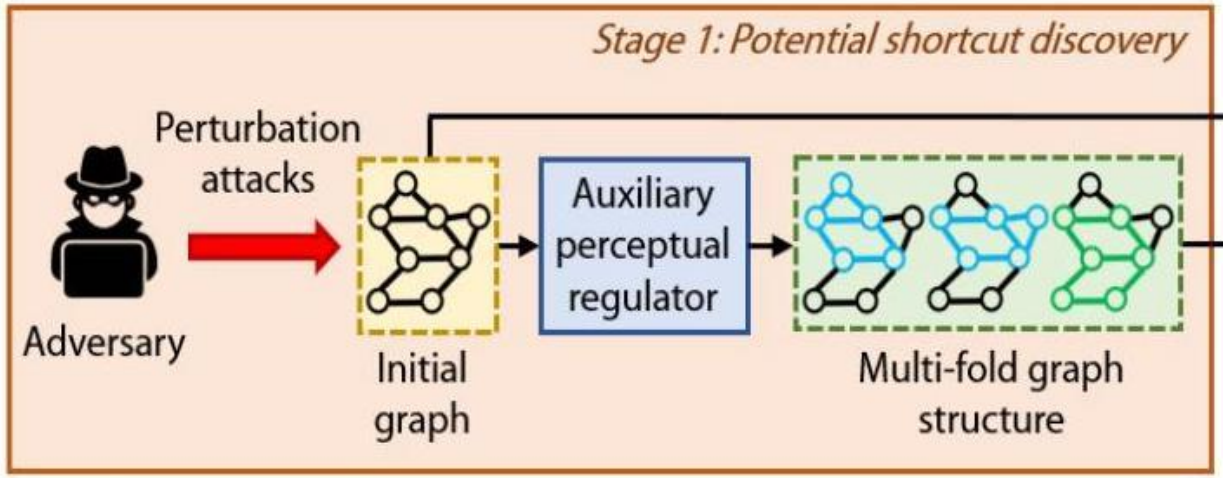
算法原理

- 算法原理图
  - 潜在捷径发现
    - 通过辅助感知调节器生成多重重叠的图结构，增强数据集的多样性并探索捷径之间的相互作用
  - 因果对齐合理化构建
    - 原始数据和增强数据通过最小化经验风险参与因果原理的构建
  - 一致性驱动的最优合理化
    - 设计因果查询、干预查询和查询反馈机制调节
    - 鼓励因果查询并抑制干预查询，进行双层协同优化过程



算法原理 潜在捷径发现

- 模块概述
  - 目标：识别与建模图神经网络训练过程中的潜在伪因果捷径依赖，并以多扰动环境模拟手段引导模型提升对因果稳定结构的敏感性
  - 输入：
    - 原始图结构数据、对应图标签、图扰动次数
  - 操作：
    - 训练多个辅助感知调节器
    - 生成多组扰动版本图
  - 输出：
    - 关于原始图结构的多组扰动版本图



算法原理 潜在捷径发现

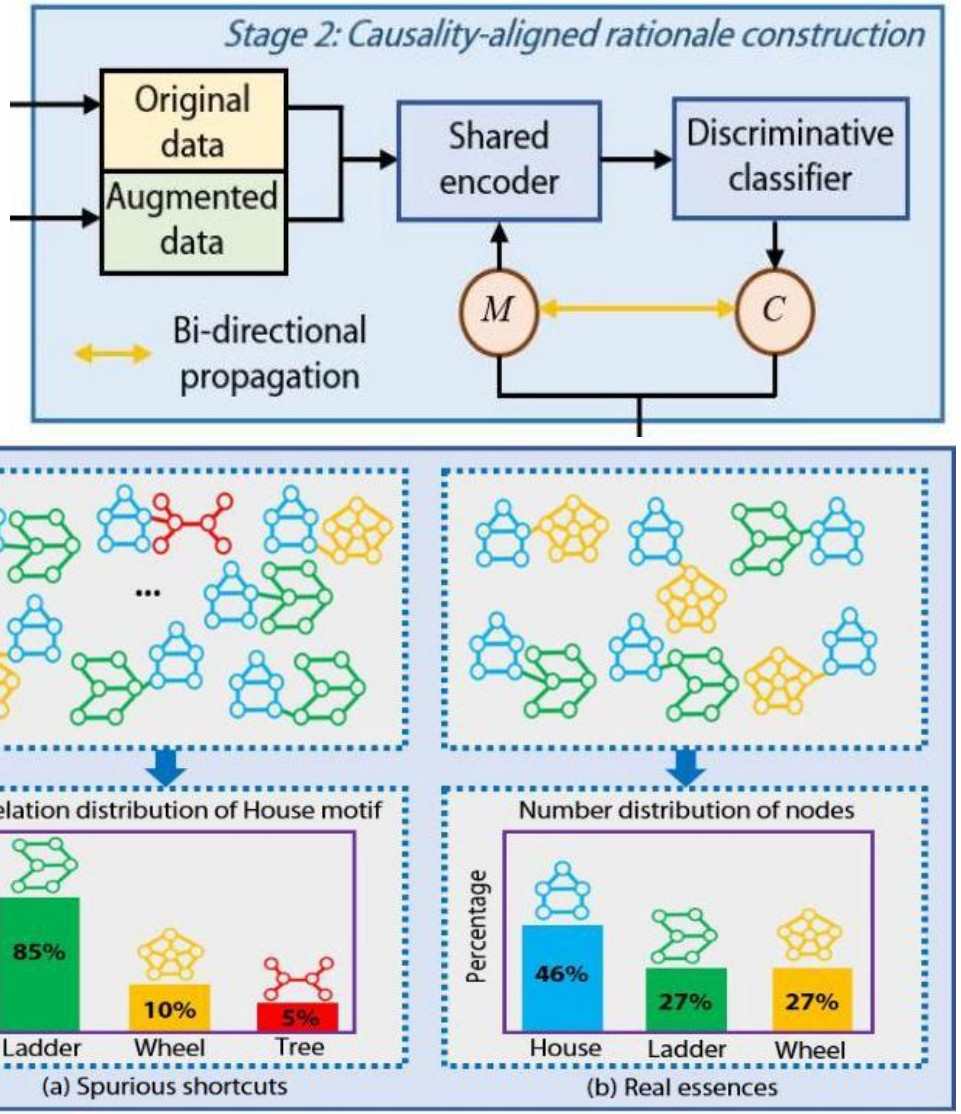
- 辅助感知调节器
  - 作用：拉大环境差异，迫使模型学习不同环境都存在的因果成分，暴露出不稳定的非因果捷径
  - 实现方法：增边、删边、轻度重连
  - 双层优化目标
    - 外层：同时最小化各环境损失的方差+ 最小化各环境损失的均值
    - 内层：给定当前 GNN，最大化跨环境损失方差，让不稳定的“捷径信号”更容易被识别并在外层被压制

$$\min_{\theta} \ \mathrm{Var}[L(r_{w_z^*}(G), Y; \theta)] \ + \ \alpha \sum_{z=1}^{Z} L(r_{w_z^*}(G), Y; \theta)$$

$$\text{s.t. } [w_1^*, \ldots, w_Z^*] = \arg\max_{w_1,\ldots,w_Z} \ \mathrm{Var}[L(r_{w_z}(G), Y; \theta)]$$

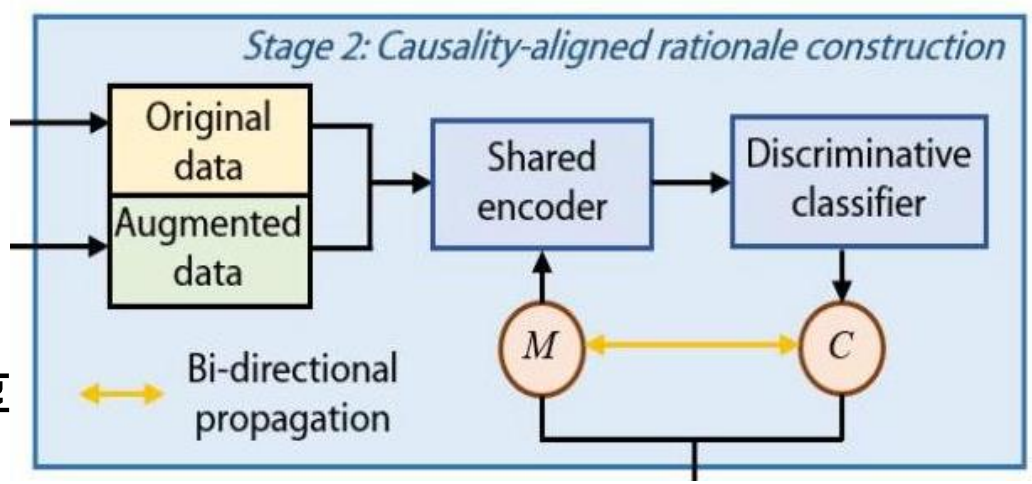
算法原理 因果对齐合理化构建

- 模块概述
  - 目标：
    - 借助上一步产生的多个环境，学习出稳定的因果特征
  - 输入：
    - 多组扰动版本图（不同环境）、GNN
  - 操作：
    - 多组扰动版本图构成多环境图数据集
    - 通过双向传播学习不变性
  - 输出：
    - 所有干预中都稳定的因果特征，即合理子图结构集



算法原理 因果对齐合理化构建

- GNN模型分解
  - 将模型h分解为用于共识表示的共享编码器f_1和用于性能预测的判别分类器f_2，即$h = f_1 \circ f_2$
  - $f_1: G \to \tilde{C}$指导从观察到的G中发现因果特征$\tilde{C}$
  - $f_2: \tilde{C} \to \hat{Y}$确保预测结果$\hat{Y}$接近$Y$



- 双向传播
  - 前向传播：原始数据和增强数据分别通过共享编码器f_1和判别分类器f_2
  - 反向传播：f_1的参数被引导至一个方向，使得其输出的$\tilde{C}$能够同时最小化所有环境的损失，并且这些损失值彼此接近

算法原理 因果对齐合理化构建

- 因果对齐
  - 原理
    - 普通经验风险最小优化目标：

$$\min_\theta \mathbb{E}_{(G,Y)}[\,l(f_2(f_1(G)),Y)\,]$$

在跨环境时有时会利用M的特征预测Y，因为训练中M可能比C更容易拟合，导致模型看似收敛、实则过拟合捷径

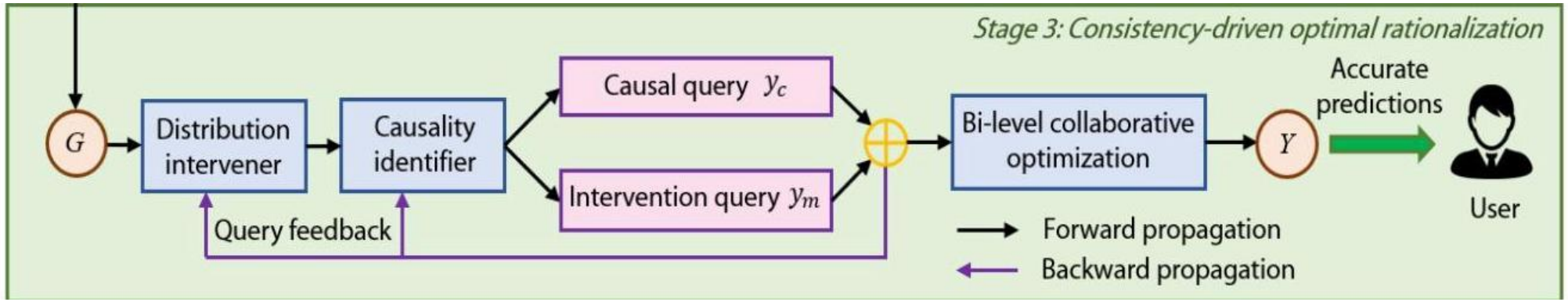
    - 因果对齐利用结构因果模型中思想，执行$do(M=m)$，即人为固定M到某个取值m，去观察Y的分布如何变化，通过干预来打破M与Y的错误联系
  - 优化目标

$$\min_\theta \ \mathbb{E}_{m\sim p(M)}[\,\mathcal{R}_m(f_1,f_2)\,]+\beta\,\mathrm{Var}_{m\sim p(M)}[\,\mathcal{R}_m(f_1,f_2)\,]$$

    - $\mathcal{R}_m(f_1,f_2)$表示在执行干预$do(M=m)$下的损失；$E_m[\cdot]$指损失的期望；$\boldsymbol{Var_m}[\cdot]$指损失的方差；$\beta$控制两者的权衡

算法原理 一致性驱动的最优合理化

- 模块概述
  - 目标：通过训练动态验证与筛选有效干预和最优解释结构
  - 输入：多组扰动版本图、合理子图结构集、GNN模型
  - 操作：双层优化
    - 内层优化当前干预下训练模型
    - 外层评估并选择下一次干预
  - 输出：最终预测模型



算法原理 一致性驱动的最优合理化

- 内层
  - 因果识别器：抽取候选因果特征$\tilde{C}$
  - 因果查询y_c和干预查询y_m：分别代表干预前后的预测结果
  - 优化目标

$$\min_{\gamma,\eta,\mu} \mathbb{E}[\mathcal{R}(h(G),Y|do(M=m))]+\beta \mathrm{Var}[\mathcal{R}(h(G),Y|do(M=m))]$$
$$+\mathbb{E}[l(\hat{\mathbf{y}}_v^m,\mathbf{y}_v)],\ \text{s.t. } Y \perp\!\!\!\perp \tilde{M} \mid \tilde{C},\ \tilde{C}=f_1(G),\ \hat{Y}=f_2(\tilde{C})$$

    前半部分和第二阶段相同，后半部分损失会完全传播给f_2，与其他组件分离，保证不影响因果关系的判断
  - 目标：固定干预下时，优化模型的性能

算法原理 一致性驱动的最优合理化

- 外层
  - 目标：主动挑选最能提升因果特征质量的干预$do(M = m)$，并由此优化模型
  - 方法：引入一个查询反馈$K = Q(\tilde{C})$得到$\tilde{C}$的不确定性，并主动选择信息增益最高的干预m，让模型在最能揭露不稳定性的场景中继续训练
  - 依据准则

$$\max_{do(M=m_\tau)} I(K, \mathbf{G}^\tau | \mathbf{G}_v^{1:\tau-1})$$

其中G^τ表示在本轮干预下采样到的数据，$G_v^{1:\tau-1}$表示迭代到$\tau - 1$轮时收集的数据集，论文中进一步将公式变形为$U(m)$，更新参数时则参考以下公式

$$M^* \in \arg\min_M U(M, G_M^*), \forall M, \text{ s.t. } G_M^* \in \arg\min_{G_M} U(M, G_M)$$

其中G_M^*是在该干预下生成的最佳样本结构，M^*是最优干预集合，分别对应内层和外层的优化结果

实验设计

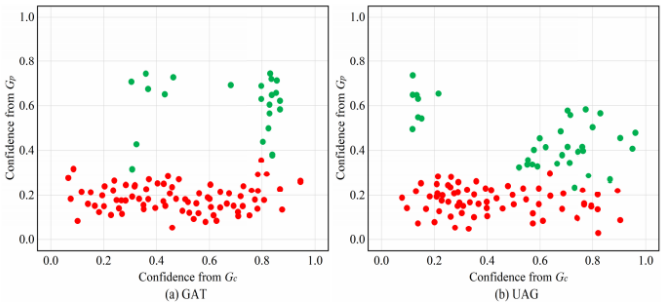
- 数据资源
  - 数据集：一个合成数据集（Spurious-Motif）+六个真实数据集
- 攻击方法
  - 随机扰动攻击（RPA）、对抗性迁移攻击（ATA）、重新布线攻击（ReWatt）、强盗优化攻击（BOA）和谱攻击（SPAC）
- 基线
  - 推理防御方法：GAT（2018）、UAG（2021）、SEP（2022）、GSN（2023）
  - 稳定学习防御方法：V-Rex（2021）、MDC（2021）、LADG（2022）
- 指标：分类准确率

Datasets	#Graphs	#Nodes	#Edges	#Classes
Spurious-Motif	18,000	25.6	35.6	3
MNIST-75sp	70,000	66.8	600.2	10
AIDS	2,000	15.7	16.2	2
NCI1	4,110	29.9	32.3	2
PC3	2,751	26.4	28.5	2
IMDB-B	1,000	19.8	193.1	2
IMDB-M	1,500	13.0	65.9	3

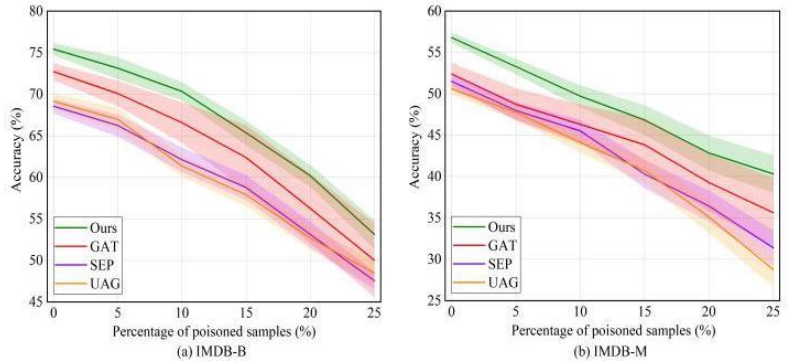
实验设计 对比实验

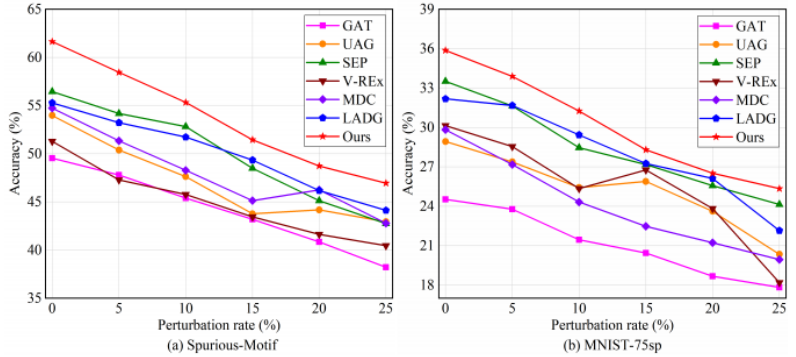
- 实验结果

Compared models	Spurious-Motif	MNIST-75sp	AIDS
GAT	45.93±6.58	19.74±4.92	94.37±4.16
UAG	50.75±3.29	23.88±7.41	95.49±2.85
SEP	52.14±7.30	28.70±2.97	96.02±3.48
V-REx	49.87±2.37	23.17±5.41	95.98±1.43
MDC	50.26±6.49	24.35±3.74	96.57±2.35
LADG	52.50±2.75	27.18±4.18	97.38±1.19
Ours	**55.83±1.86**	**29.37±2.17**	**98.15±0.93**



Compared models	Spurious-Motif			
	b = 0.6	b = 0.7	b = 0.8	b = 0.9
GAT	40.36±6.37	39.04±6.71	38.42±5.83	34.67±5.23
UAG	45.78±2.45	43.62±5.87	41.57±4.62	38.15±3.95
SEP	46.81±5.61	43.14±4.75	40.28±4.48	37.06±1.74
V-REx	43.12±3.78	41.83±2.36	38.42±3.29	35.64±3.79
MDC	44.64±2.85	42.14±1.94	37.16±3.73	36.18±2.02
LADG	45.98±4.63	41.75±3.92	38.42±1.82	38.01±2.36
Ours	**50.15±2.79**	**48.40±1.84**	**44.92±3.58**	**41.78±0.96**

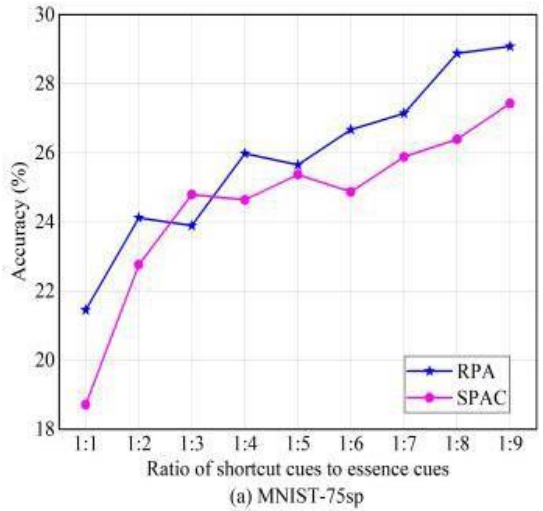




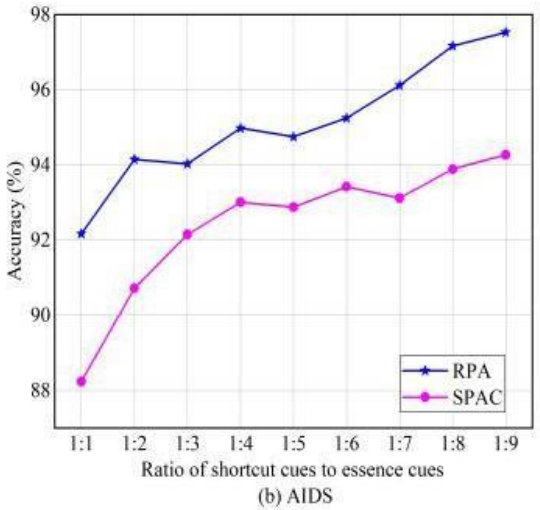
- 分析
  - CASR在面对五种不同的攻击时都表现最佳
  - 对比方法没能学习到抗干扰能力强的因果依赖关系，导致性能不如CASR

实验设计 消融实验

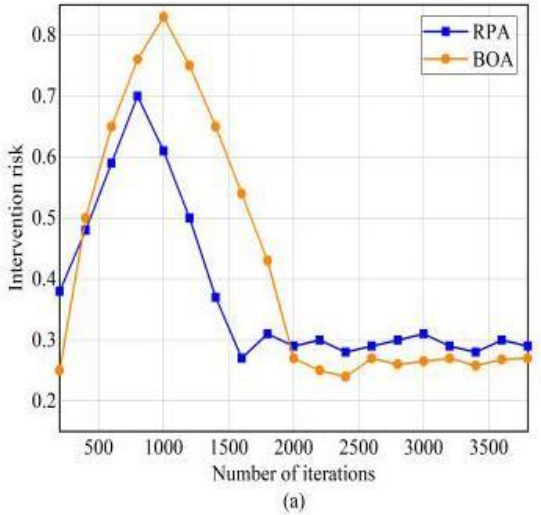
- 实验结果



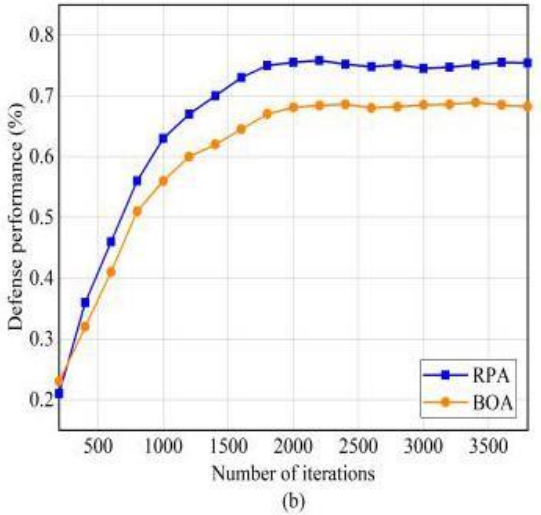
(a) MNIST-75sp



(b) AIDS



(a)



(b)

- 消融情况
  - 通过权重分配调整捷径线索与因果线索之间的比例，研究虚假相关性的影响
    - 从局部角度来看，在小范围内捷径百分比的降低不一定会导致性能提升，因为捷径线索和本质线索可以相互连接和相互作用
  - 观察在一致性驱动的合理化过程中，干预损失和防御性能随迭代次数的变化
    - 防御性能随着干预损失的增加而迅速提高，但随着干预损失的降低而增长缓慢

Toward Intelligent Attack Detection With Causal Transformer in Internet of Things

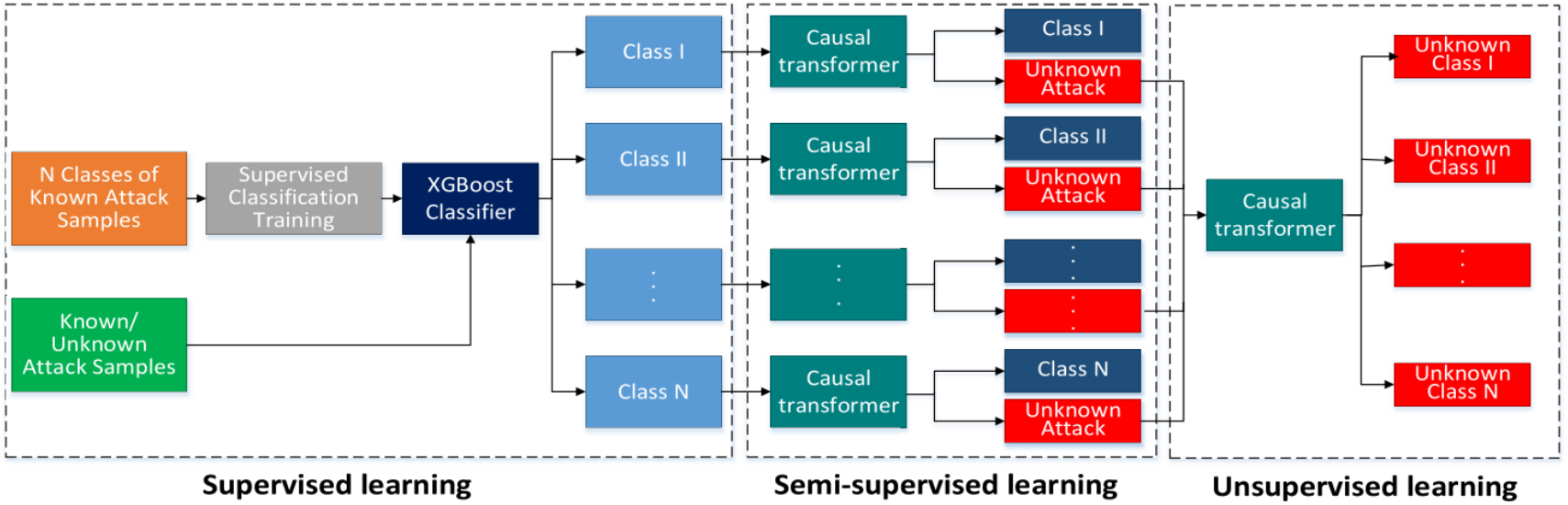
算法原理 TIPO

T	目标	的能够同时识别已知攻击与未知攻击智能物联网攻击检测系统
I	输入	物联网网络正常和攻击流量数据集（Darknet、ToN_IoT、NSL_KDD）
P	处理	1.基于因果推理提取特征独立权重与因果效应权重 2. 在Transformer注意力机制中引入因果加权，去除伪相关 3. 通过三阶段学习实现攻击识别
O	输出	高精度检测结果

P	问题	传统机器学习与Transformer在物联网入侵检测中仅依靠特征相关性判断，无法剔除噪声特征引起的伪相关，导致对未知攻击的误判率与漏报率高
C	条件	数据集包含多类型流量（正常、已知、未知）
D	难点	如何利用因果推理机制有效分离真实攻击特征与噪声特征，消除伪相关
L	水平	IEEE IoT Journal 2025 中科院2区

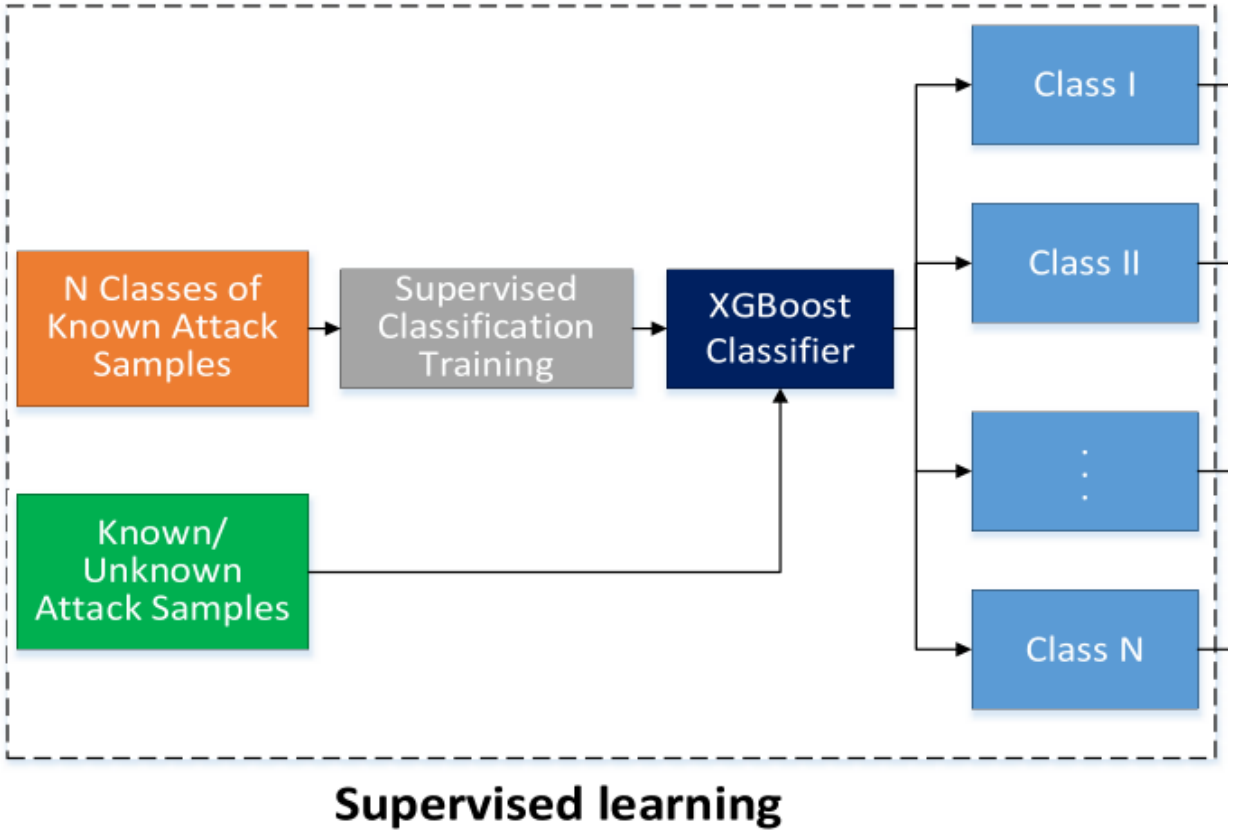
算法原理

- 算法原理图
  - 监督学习模块
    - 用于分类已知的攻击类型
  - 半监督学习模块
    - 通过因果推理，区分已知和未知的攻击类型
  - 无监督学习模块
    - 根据检测到的流量的因果特征来分类未知的攻击

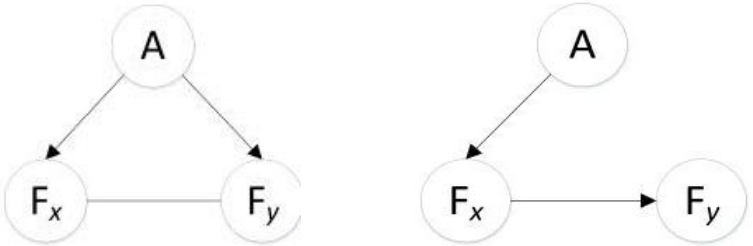


算法原理 监督学习模块

- 模块概述
  - 目标：
    - 对已知攻击进行准确分类
  - 输入：
    - 正常流量和已知攻击
  - 操作：
    - 基于因果推理训练因果效应权重
    - 根据因果效应的大小去除噪声特征
    - 基于监督XGBoost方法训练分类器
  - 输出
    - 已知攻击分类



- 因果推理
  - 因果关系：比较同一网络攻击在有干预和无干预情况下各流量特征的异常情况
  - 基于权重矩阵W的特征独立性准则：
    - 对于两个非独立的同分布特征F_x、F_y，如果存在一个$p \times 1$阶的加权矩阵 W，使得 $E\left(F_x \Sigma W F_y \middle| A\right) = E(F_x W|A)E\left(F_y W|A\right)$，其中$\Sigma W$是一个对角矩阵且 $W_{i,1} = (\Sigma W)_{i,i}$，那么$W$的存在可以$F_x$、$F_y$相互独立
  - 因果效应ICE：
    - 反转A和F之间的因果方向，并通过改变F来判断A的变化
    - 当特征异常$(F = 1)$时的平均结果与网络特征正常$(F = 0)$时的平均结果之间的差异就是F和A间的平均因果效应



- 特征权重
  - 训练得到使特征之间相互独立的样本权重　　使用变量去相关正则器
    - 减少训练环境中特征之间以及网络攻击与特征之间的虚假相关性
    - 确保网络攻击与特征之间因果关系计算的准确性和唯一性

$$G=\min_{G}\sum_{j=1}^{p}\parallel E\left(X_{j}^{0,\mathrm{T}}\sum gX_{(I-j)}^{0}\right)-E\left(X_{j}^{0,\mathrm{T}}G\right)E\left(X_{(I-j)}^{0,\mathrm{T}}G\right)\Big|_{2}^{2}$$

  - 接着训练获得反应特征和标签间因果效应的因果效应权重
    - 因果特征对标签的影响可以通过因果权重增强
    - 噪声特征与标签之间的伪相关性可以解耦

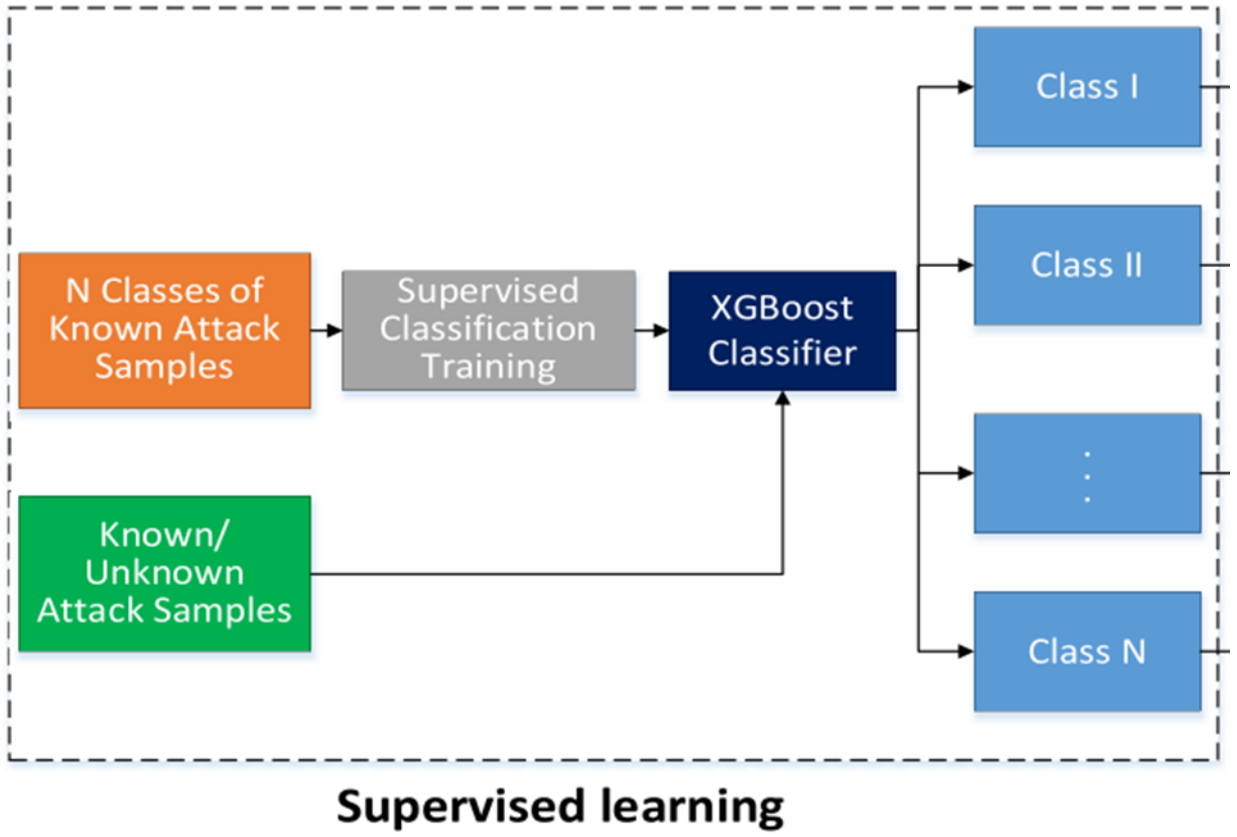
$$W=\arg\min_{W}\sum_{i=1}^{p}\parallel E\left(GF_{i}^{0}\sum wGF_{(I-i)}^{0}\right)-E\left(GF_{i}^{0}W\right)E\left(W^{\mathrm{T}}GF_{(I-i)}^{0}\right)\parallel_{2}^{2}$$

算法原理 监督学习模块

- 特征筛选
  - 根据公式剔除噪声特征F_n，保留因果特征F_c

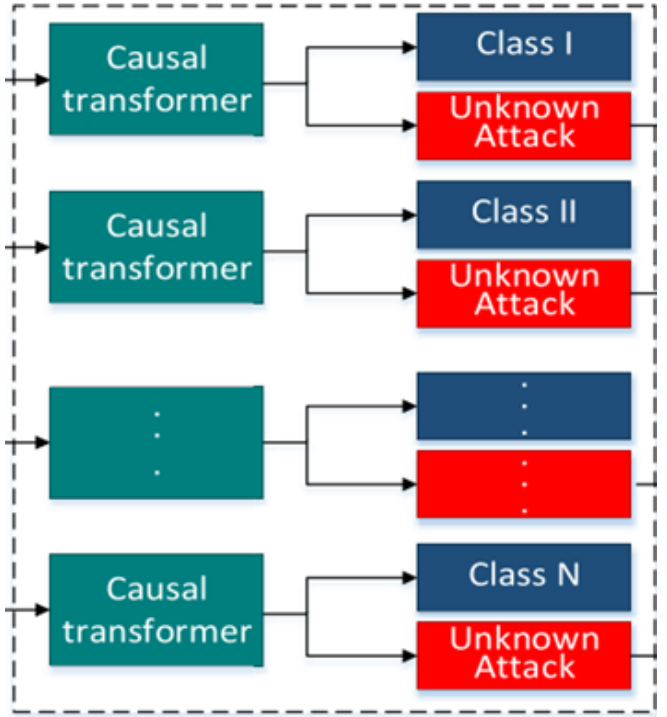
$$E\left(F_c \sum wF_n^{\mathrm{T}}\right) = E\left(F_c W^{\mathrm{T}}\right)$$

  - 根据特征集 F_c 筛选构造训练样本 x_c，其中样本特征间满足因果独立性
- 模型训练
  - 训练XGBoost分类器
  - 将要预测的样本输入，得到对攻击类型的分类



算法原理 半监督学习模块

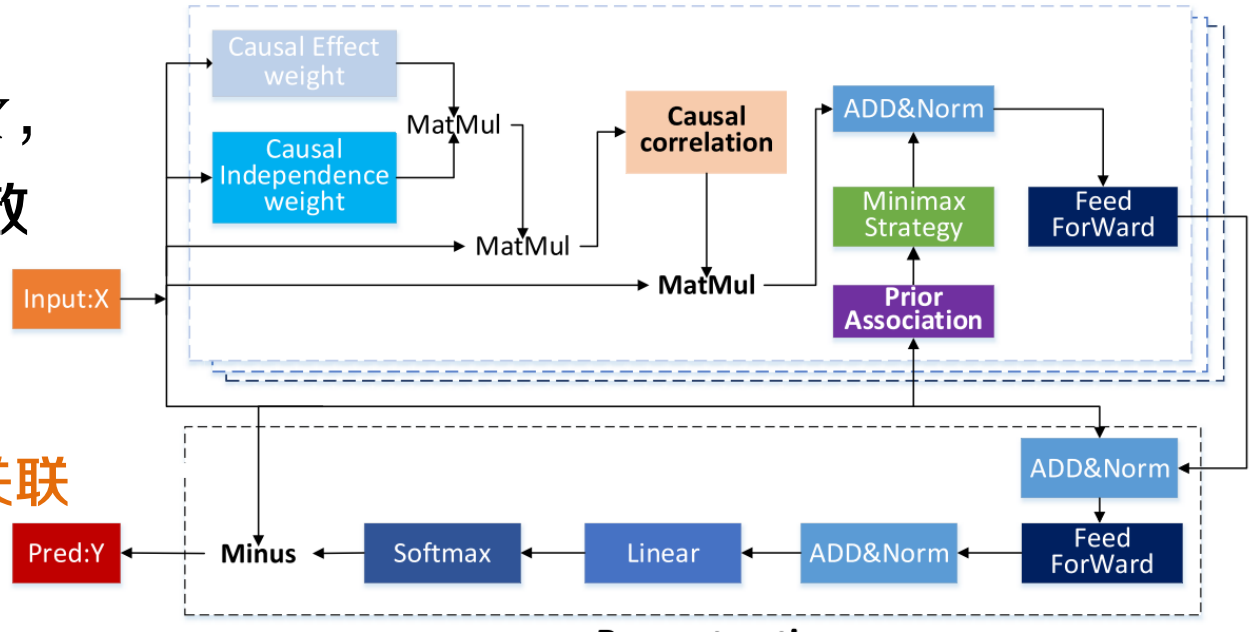
- 模块概述
  - 目标：
    - 通过因果推理，区分已知和未知的攻击类型
  - 输入：
    - 监督模块初判为该已知类（或正常）的检测样本子集
  - 操作：
    - 因果transformer
      - 极小极大策略
  - 输出
    - 跟据每个样本的异常打分，把样本标为该已知类的内点或外点
    - 识别为未知攻击类型的样本组成统一集合



Semi-supervised learning

算法原理 半监督学习模块

- 因果transformer
  - 把原Query改为$Q = G^T W$； $K = V = X$，让模型关注输入序列中具有更大因果效应的部分，同时忽略虚假相关性
  - 计算因果注意力CA和先验关联PA
    - 通过因果序列注意力分布学习因果关联
    - 用高斯核在特征维上构造先验关联
  - ADis与极小极大策略
    - ADis：度量CA和PA的差异
    - 极小极大策略：

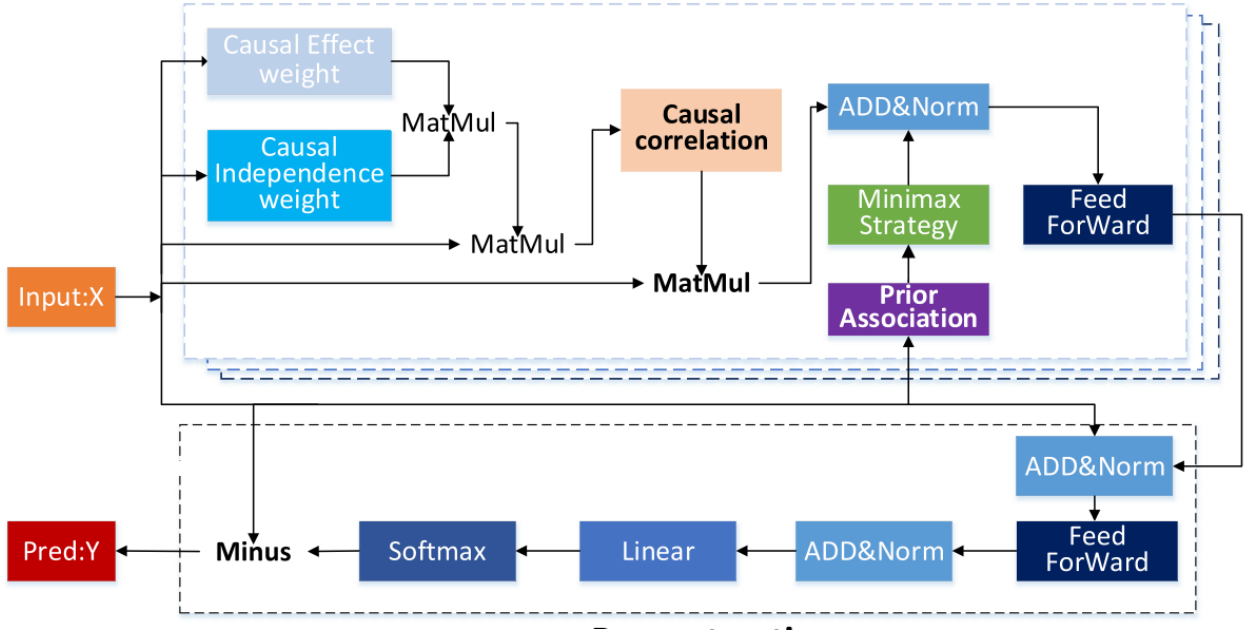


本类样本的重构误差被压低，伪相关样本的误差被拉高

  - CA的反向传播停止：优化先验关联PA，使先验关联近似从原始样本中学到的 CA
  - CA的优化停止：使CA更加关注具有最大因果效应和标签的特征

算法原理 半监督学习模块

- 因果transformer
  - 重构样本并比较误差
    - Add & Norm + Feed Forward
    - 使用欧式距离计算重构误差
  - 异常度评分
    - 通过Softmax对$ADis$进行归一化
    - 最终的异常分数Pred：



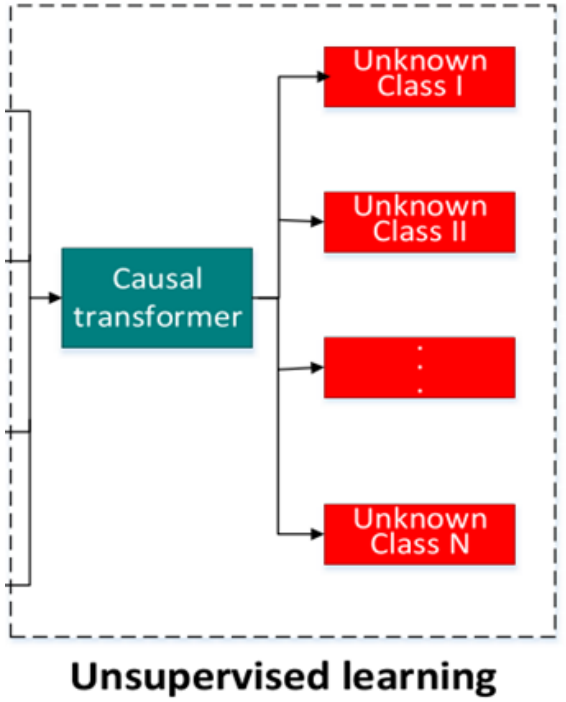
$$\text{pred} = ||\text{GW}||_2^2 \text{Softmax}(-\text{ADis}) \odot \sum_{i=1}^{p} ||F_{i,:} - \text{outF2}_{i,:}||_2^2$$

其中$\|GW\|_2^2$代表因果效应的强度；Softmax(-ADis)指根据因果关联的差异加权；

$\left\|F_{i,:} - outF2_{i,:}\right\|_2^2$代表重构误差，表示样本的异常程度

算法原理 无监督学习模块

- 模块概述
  - 目标：
    - 把半监督模块判定为外点的样本进一步分组，归为若干未知攻击类型
  - 输入：
    - 来自各个半监督检测器被判为未知的样本集合
  - 操作：
    - 因果transformer
    - 样本聚类和拆分
  - 输出
    - 样本分成若干未知类簇的分组结果



Unsupervised learning

$X = \{UA_i\}_{i=1,2,...,C}$//A total of C causal transformer detectors
$Z.append(X)$;$UC = \varnothing$;$pred = [0]*m$// UC is a sample of an unknown attack type that has been classified
for ln in range ($len(Z)$) **do**
  $X = Z[\text{ln}]$;$m =$ $X.shape[1]$; $p = X.shape[1]$;
  Calculate $\mathcal{L}_G$ according to Equation (27);
  Calculate $\mathcal{L}_W$ according to Equation (28);
  $Q = G^T W$;
  $K = V = X$;
  for (Q_l, K_l, V_l, X^l) in (Q, K, V, X): **do**
    $CA_{l-1}(Q_{l-1}, K_{l-1}, V_{l-1})$ and PA_{l-1} are calculated according to Equations (18)-(19).
    Calculate $out1$ according to Equation (30);
    Calculate $out2$ according to Equation (31);
    $X^l = out2$;
    The sample X^l is recharacterized according to Equation (29);
  end for
  The reconstruction error is calculated according to Equation (32), and $pred$ is obtained;
  if ($|pred_0 - pred_1| < \delta$)// The reconstruction error is less than the threshold. **then**
    $UC.append(X^l)$, $Z.Delete(X^l)$;// Add to the set of already classified unknown types, and remove from Z
  else
    $Z.ppend(X_{pred_1})$ // Add to Z as a new set.
    $ln = ln - 1$
  end if
end for
return$Z$

算法原理 无监督学习模块

- 因果transformer
  - 和半监督模块过程相同，获得pred分数
- 聚类决策
  - 从待分类样本集Z中依次取出样本进行处理
  - 阈值比较

$$|pred_0 - pred_1| < \delta$$

    - 样本子集内部重构误差差异很小，则加入分类集合，将样本从Z删去
    - 若差异较大，将样本返回Z，等待下一轮处理
    - 所有样本经过一次两两比较后，聚类过程就停止
  - 最终所有的未知样本被分类成不同的簇

$X = \{UA_i\}_{i=1,2,...,C}$//A total of C causal transformer detectors
$Z.append(X)$;$UC = \varnothing$;$pred = [0]*m$// UC is a sample of an unknown attack type that has been classified
for ln in range ($len(Z)$) **do**
  $X = Z[\text{ln}]$;$m = X.shape[1]$; $p = X.shape[1]$;
  Calculate $\mathcal{L}_G$ according to Equation (27);
  Calculate $\mathcal{L}_W$ according to Equation (28);
  $Q = G^T W$;
  $K = V = X$;
  for (Q_l, K_l, V_l, X^l) in (Q, K, V, X): **do**
    $CA_{l-1}(Q_{l-1}, K_{l-1}, V_{l-1})$ and PA_{l-1} are calculated according to Equations (18)-(19).
    Calculate $out1$ according to Equation (30);
    Calculate $out2$ according to Equation (31);
    $X^l = out2$;
    The sample X^l is recharacterized according to Equation (29);
  end for
  The reconstruction error is calculated according to Equation (32), and $pred$ is obtained;
  if ($|pred_0 - pred_1| < \delta$)// The reconstruction error is less than the threshold. **then**
    $UC.append(X^l)$, $Z.Delete(X^l)$;// Add to the set of already classified unknown types, and remove from Z
  else
    $Z.ppend(X_{pred_1})$ // Add to Z as a new set.
    $ln = ln - 1$
  end if
end for
return$Z$

实验设计

- 数据资源
  - 数据集：Darknet（样本数：31,192）、ToN_IoT（样本数：8430）和NSL_KDD（样本数：14726）
- 基线
  - 传统机器学习方法：Isolation Forest（2022）、SVM（2013）、LOF（2022）
  - 循环神经网络方法：LSTM_Univariate（2018）、LSTM_AD（2022）、LSTM（2021）
  - 生成模型与自编码器方法：DAGMM（2024）、OmniAnomaly（2022）、USAD（2020）、MAD_GAN（2019）
  - Transformer与注意力方法：TranAD（2022）、Attention（2019）
  - 其他深度学习方法：MSCRED（2019）、MTAD_GAT（2023）、GDN（2021）
- 指标：Accuracy、Precision、F1-score

实验设计 对比实验

- 实验结果

Methods	Non-Tor			Non-VPN			Tor		
	Ac	Pr	F1	Ac	Pr	F1	Ac	Pr	F1
TranAD	0.9540	0.4968	0.5723	0.9069	0.0905	0.0404	0.9793	0.6946	0.8198
LSTM_Univariate	0.9571	0.5238	0.5891	0.9730	0.7359	0.8479	0.9711	0.6202	0.7656
Attention	0.9549	0.5046	0.5768	0.9723	0.7315	0.8449	0.9660	0.5810	0.7350
LSTM_AD	0.9547	0.5026	0.5774	0.9722	0.7307	0.8444	0.9700	0.6113	0.7587
DAGMM	0.9505	0.4709	0.5559	0.9740	0.7434	0.8528	0.9845	0.7527	0.8589
OmniAnomaly	0.9544	0.5006	0.5748	0.9715	0.7256	0.8410	0.9728	0.6342	0.7761
USAD	0.9555	0.5092	0.5823	0.9715	0.7256	0.8410	0.9724	0.6304	0.7733
MSCRED	0.9523	0.4837	0.5635	0.9732	0.7374	0.8488	0.9824	0.7281	0.8426
MTAD_GAT	0.9519	0.4806	0.5602	0.9720	0.7293	0.8434	0.9757	0.6599	0.7951
GDN	0.9542	0.4987	0.5723	0.9711	0.7228	0.8391	0.9754	0.6568	0.7929
MAD_GAN	0.9543	0.5000	0.5744	0.9723	0.7315	0.8449	0.9756	0.6589	0.7944
Isolation Forest	0.7521	0.0230	0.0378	0.8464	0.1877	0.2344	0.8780	0.1655	0.2328
SVM	0.9449	0.1648	0.0775	0.8342	0.0602	0.0695	0.9223	0.0772	0.0669
LOF	0.8955	0.0353	0.0410	0.6956	0.0741	0.1157	0.9216	0.0858	0.0762
LSTM	0.6051	0.0268	0.0477	0.5228	0.0920	0.1595	0.4440	0.0638	0.1181
Causal Transformer	**0.9848**	**1.0000**	**0.8008**	**1.0000**	**1.0000**	**1.0000**	**0.9998**	**0.9953**	**0.9976**

Methods	Normal			backdoor			ddos			password		
	Ac	Pr	F1	Ac	Pr	F1	Ac	Pr	F1	Ac	Pr	F1
TranAD	0.9471	0.3801	0.4571	0.9878	0.9711	0.9853	0.9876	0.9639	0.9813	0.9772	0.9653	0.9813
LSTM_Univariate	0.9581	0.4575	0.6382	0.99	0.9762	0.988	0.9914	0.9749	0.9869	0.9348	0.9673	0.9869
Attention	0.9622	0.433	0.6667	0.9884	0.9726	0.9861	0.9888	0.9673	0.983	0.9267	0.9785	0.983
LSTM_AD	0.9602	0.4765	0.6768	0.9904	0.9772	0.9885	0.9924	0.9777	0.9884	0.9876	0.9766	0.9884
DAGMM	0.9553	0.4128	0.5865	0.9907	0.9777	0.9887	0.9893	0.9687	0.9837	0.9783	0.9777	0.9837
OmniAnomaly	0.9545	0.4501	0.5765	0.9918	0.9803	0.9901	0.991	0.9735	0.9862	0.987	0.9623	0.9862
USAD	0.9708	0.4171	0.6127	0.9889	0.9736	0.9866	0.9893	0.9687	0.9837	0.9792	0.9681	0.9837
MSCRED	0.9702	0.5282	0.7132	0.9882	0.9721	0.9858	0.9792	0.5385	0.7121	0.9692	0.6385	0.7452
MTAD_GAT	0.9608	0.4471	0.6189	0.9922	0.9814	0.9906	0.9917	0.9756	0.9873	0.9873	0.8752	0.9873
GDN	0.9802	0.5281	0.7143	0.9891	0.9741	0.9869	0.9888	0.9673	0.983	0.9898	0.9572	0.983
MAD_GAN	0.9649	0.33	0.5585	0.9902	0.9767	0.9882	0.9352	0.9739	0.8917	0.9632	0.9841	0.8917
Isolation Forest	0.7589	0.1886	0.3028	0.5043	0.2221	0.1208	0.5174	0.1234	0.0971	0.613	0.2222	0.0971
SVM	0.8746	0.3367	0.3421	0.5982	0.5319	0.2647	0.6614	0.4342	0.2178	0.65	0.4332	0.2178
LOF	0.9267	0.3452	0.3322	0.576	0.368	0.0819	0.6493	0.1509	0.0316	0.6561	0.251	0.0316
LSTM	0.791	0.1229	0.235	0.5164	0.3689	0.2987	0.5202	0.2833	0.2977	0.6202	0.2933	0.2977
Causal Transformer	**1**	**1**	**1**	**0.998**	**0.9951**	**0.9975**	**0.9964**	**0.9894**	**0.9943**	**0.9932**	**0.9865**	**0.9943**

实验设计 对比实验

- 实验结果

Methods	ransomware			scanning			xss		
	Ac	Pr	F1	Ac	Pr	F1	Ac	Pr	F1
TranAD	0.9876	0.9639	0.9815	0.9657	0.9631	0.9776	0	0	0
LSTM_Univariate	0.9914	0.9749	0.9346	0.9321	0.9743	0.9880	0	0	0
Attention	0.9888	0.9673	0.9833	0.9111	0.9635	0.9801	0	0	0
LSTM_AD	0.9924	0.9777	0.9856	0.9651	0.9726	0.9817	0	0	0
DAGMM	0.9893	0.9687	0.9651	0.9876	0.9769	0.9838	0	0	0
OmniAnomaly	0.991	0.9735	0.9732	0.9456	0.9792	0.9846	0	0	0
USAD	0.9893	0.9687	0.968	0.9888	0.9725	0.9822	0	0	0
MSCRED	0.9792	0.5385	0.7312	0.9875	0.9707	0.9849	0	0	0
MTAD_GAT	0.9917	0.9756	0.9556	0.9931	0.9778	0.9825	0	0	0
GDN	0.9888	0.9673	0.9781	0.9777	0.9708	0.9774	0	0	0
MAD_GAN	0.9352	0.9739	0.8897	0.9675	0.9747	0.9818	0	0	0
Isolation Forest	0.5174	0.1234	0.1972	0.612	0.2197	0.1117	0	0	0
SVM	0.6614	0.4342	0.3172	0.5899	0.5309	0.2585	0	0	0
LOF	0.6493	0.1509	0.1315	0.5991	0.3587	0.0765	0	0	0
LSTM	0.5202	0.2833	0.3214	0.6132	0.3662	0.2943	0	0	0
Causal Transformer	**0.9964**	**0.9894**	**0.9932**	**0.997**	**0.9916**	**0.9974**	0	0	0

- 分析
  - 算法在三个数据集上区分攻击种类效果最好
  - 算法通过因果注意力和极小极大策略进行优化，消除了虚假关联，能够从已知流量类型中准确区分出不同的流量类型，而且也能有效分类未知攻击

实验设计 对比实验

- 实验结果

Methods	Ac	Pr	F1
TranAD	0.9774	0.9646	0.9820
LSTM_Univariate	0.9798	0.9683	0.9839
Attention	0.9809	0.9700	0.9848
LSTM_AD	0.9857	0.9773	0.9885
DAGMM	0.9836	0.9741	0.9869
OmniAnomaly	0.9851	0.9764	0.9881
USAD	0.9742	0.9599	0.9795
MSCRED	0.9847	0.9758	0.9878
MTAD_GAT	0.9798	0.9682	0.9839
GDN	0.9849	0.9761	0.9879
MAD_GAN	0.9815	0.9709	0.9852
Causal transformer	**0.9950**	**0.9919**	**0.9959**

Methods	Ac	Pr	F1
TranAD	0.6983	0.6989	0.6994
LSTM_Univariate	0.6993	0.6997	0.6995
Attention	0.6992	0.6994	0.6955
LSTM_AD	0.8546	0.8546	0.8546
DAGMM	0.7994	0.7996	0.7998
OmniAnomaly	0.7987	0.7983	0.7992
USAD	0.8241	0.8241	0.8241
MSCRED	0.6532	0.6532	0.6532
MTAD_GAT	0.7516	0.7516	0.7516
GDN	0.7516	0.7516	0.7516
MAD_GAN	0.5546	0.5546	0.5546
Causal Transformer	**0.9890**	**0.9890**	**0.9890**

Methods	Ac	Pr	F1
TranAD	0.4790	0.9992	0.6328
LSTM_Univariate	0.5717	0.9992	0.7167
Attention	0.5131	0.9995	0.6648
LSTM_AD	0.5049	0.9991	0.6574
DAGMM	0.5278	0.9994	0.6783
OmniAnomaly	0.5495	0.9991	0.6976
USAD	0.5696	0.9993	0.7149
MSCRED	0.5066	0.9993	0.6589
MTAD_GAT	0.5872	0.9994	0.7297
GDN	0.5034	0.9991	0.6560
MAD_GAN	0.3829	0.9994	0.5334
Causal Transformer	**0.9972**	**0.9998**	**0.9985**

- 分析
  - 算法在三个数据集上异常检测分数均为最优
  - 对正常流量样本进行训练后，获得了所有数据集正常流量样本的因果特征，并减小了与正常流量类型无因果关联的特征的影响

特点总结与未来展望

工作总结

特点总结与工作展望

- 特点总结
  - CASR
    - 通过因果对齐来抑制、去除伪相关捷径并强化因果特征
    - 使用查询反馈与双层优化实现
  - CTIoT
    - 用 do-干预学习样本权重G使特征近似独立，再为各特征学习因果效应权重W，减小特征之间干扰，增加辨识能力
    - 使用因果注意力，为不同特征增加相应权重，更容易提取攻击信息，抑制伪相关
- 工作展望
  - 降低捷径比例在局部区间并不一定提升性能，二者可能相互作用，需要平衡决策
  - 对于极端不平衡的数据难以处理，可以单独处理数据样本较少的类别，进行主动填充数据等

参考文献

[1] Jia J, Ma S, Liu Y, et al. A causality-aligned structure rationalization scheme against adversarial biased perturbations for graph neural networks[J]. IEEE Transactions on Information Forensics and Security, 2024.

[2] Zeng Z, Zhao B, Deng X, et al. Toward intelligent attack detection with causal transformer in Internet of Things[J]. IEEE Internet of Things Journal, 2025.

道德经

知人者智，自知者明。胜人者有力，自胜者强。知足者富。强行者有志。不失其所者久。死而不亡者，寿。

谢谢！