Beijing Forest Studio

北京理工大学信息系统及安全对抗实验中心

深度学习模型公平性修复

硕士研究生　张耕源

2025 年 09 月 07 日

问题回溯

- 相关内容
  - 2025.03.30 刘洧光 《人工智能模型的公平性测试——既要公平，也要正确》
  - 2024.09.28 刘洧光 《人工智能模型的公平性测试》
  - 2022.08.23 王若辉《AI测试：历史与发展》

背景简介

- 预期收获
  - 掌握深度学习模型公平性修复相关知识
  - 了解一种深度学习模型公平性修复方法

内涵解析与研究目标

- 题目内涵解析（深度学习模型公平性修复技术研究）
  - 公平性：同一时间对于所有人平等对待的性质，是模型重要的非功能性需求之一
  - 公平性修复：识别和修复模型中的不公平因素，使模型在相似个体和不同群体之间决策公正
  - 群体公平性：模型在不同敏感属性群体之间结果平等

- 研究目标
  - 面向深度学习模型公平性修复
  - 研究数据集公平、算法公平、模型公平等关键问题
  - 研究低准确率衰减的模型公平性修复

研究背景与实际意义

- 研究背景
  - 深度学习模型具有强大的特征提取能力，在决策领域得到广泛应用，但往往产生不公平的预测结果，造成不良的社会影响
- 研究意义
  - 缓解深度神经网络（DNN）在高风险应用中存在的公平性问题

谷歌翻译是世界上最受欢迎的翻译引擎,它显示出性别偏见。"她是工程师,他是护士"翻译成土耳其语后,再翻译成英语,就变成了"他是工程师,她是护士"

一款广泛使用的人脸识别软件被发现对深肤色女性有偏见，对于深肤色女性识别错误率更高

研究历史

Udeshi等人提出Aequitas，这是首个基于两阶段搜索框架的公平性测试方法。在全局搜索阶段，Aequitas随机探索输入空间以发现歧视性实例；在局部搜索阶段，通过扰动全局阶段找到的歧视性实例的非敏感属性，探索其邻近输入以识别更多歧视性实例

Sharma等人提出FairCheck和MLCheck，通过用白盒模型近似黑盒测试模型，利用SMT求解器将公平性属性和白盒模型转换为逻辑公式，自动生成试图违反指定公平性属性的测试用例

Zhang等人提出提出了EIDIG框架，结合了快速生成一组多样化的判别种子的全局生成阶段和梯度指导下在这些种子周围生成尽可能多的个体判别实例的局部生成阶段，实验表明该算法在有效性和效率上均优于现有方法

Li等人假设受保护属性相关特征会影响深度神经网络的公平性，提出Faire方法，通过神经元条件合成来降低受保护属性特征的影响，修复深度神经网络的个体歧视问题

Adel等人假设当自动化决策系统的结果受敏感属性影响，对不同敏感属性子群体产生不同对待时就存在不公平性，提出仅对现有可微分类器架构添加新隐藏层和新分类器，利用单网络对抗学习框架优化公平性

Zhang等人提出了一种基于梯度的可扩展算法对抗歧视查找器，通过对抗采样为DNN生成个体歧视性实例，实验表明该算法在有效性和效率上均优于现有方法

Gao等人提出了Fairneuron，通过神经元切片识别冲突路径，样本聚类得到普通样本和有偏样本，对两种样本采取不同的训练策略实现低准确率衰减的公平性修复

Li等人提出了RUNNER，设计了基于重要性的神经元诊断，利用新的神经元重要性标准一步识别出负责任的不公平神经元；通过增加损失项来测量所有来源中不同子群的负责不公平神经元的激活距离，从而设计神经元稳定再训练

知识基础 模型公平性指标（群体公平性）

- 人口统计学均等（ΔDP）
  - 计算公式

$$\Delta DP = |P(\hat{y} = 1|a = 0) - P(\hat{y} = 1|a = 1)|$$

  - 公式说明
    - $\hat{y}$为模型预测结果，$\hat{y} = 1$代表有利的预测结果（模型判定的“阳性类别”，如“贷款通过”）
    - a为敏感属性，对于弱势群体，a设置为0
  - 场景示例
    - 男性贷款获批率0.8，女性0.6，则$\Delta DP = 0.2$，模型存在歧视
  - 特点
    - 只看“模型预测结果”，不看“样本的真实情况”
    - 计算简单但是可能出现“公平但预测质量差”的情况

知识基础 模型公平性指标（群体公平性）

- 人口统计学均等（ΔEO）
  - 计算公式

$$\Delta EO = |P(\hat{y} = 1|a = 0, y = 1) - P(\hat{y} = 1|a = 1, y = 1)| + |P(\hat{y} = 1|a = 0, y = 0) - P(\hat{y} = 1|a = 1, y = 0)|$$

  - 公式说明
    - $\hat{y}$为模型预测结果，$\hat{y} = 1$代表有利的预测结果，$y = 1$代表真实为阳性
    - 公式本质为ΔEO =|真阳性率差异|+|假阳性率差异|
  - 场景示例
    - 比如“真实收入> 50K的男性”和“真实收入> 50K的女性”，被模型预测为“收入> 50K”的概率应相等；“真实收入≤50K的男性”和“真实收入≤50K的女性”，被模型误判为“收入> 50K”的概率也应相等。
  - 特点
    - 考虑真实标签，避免因公平而牺牲准确率，更符合“公平直觉”

知识基础 模型准确性指标（AP）

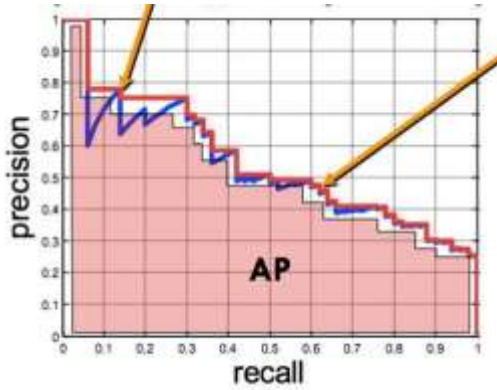
- 准确率Precision

$$Precision = \frac{TP}{TP + FP}$$

  - TP为正例预测为正，FP为负例预测为正
- 召回率Recall

$$Recall = \frac{TP}{TP + FN}$$

  - FN为正例预测为负



Average Precision (AP)
0% is worst
100% is best

mean AP over classes (mAP)

- 精度-召回曲线（PR曲线）
  - 将Precision（精度）和Recall（召回率）的关系绘制成一张曲线图。通过设置不同的阈值影响TP、FP、FN的值，计算出不同阈值下的Precision和Recall值，所有阈值对应的精确率（Y轴）和召回率（X轴）绘制到图中，形成PR曲线。

知识基础 一阶泰勒变换

- 一阶泰勒变换
  - 计算公式

$$f(x) \approx f(x_0) + f'(x_0) \cdot (x - x_0)$$

  - 公式说明
    - $f(x_0)$：函数在x_0处的实际值
    - $f'(x_0)$：函数在x_0处的以及导数（$f(x)$可微）
    - $(x - x_0)$：自变量x与参考点的偏差
  - 核心逻辑
    - 用参考点的函数值+偏差的线性修正项，近似描述x附近的函数行为

知识基础 top-k和L1距离

- ***top-k***
  - 计算公式

$$\Omega_k^l = topk(\omega l, UI^l)$$

  - 公式说明
    - ωl：第l层的神经元
    - $topk(\omega l, UI^l)$：每一层中神经元不公平重要性前k的神经元
- ***L1*距离**
  - 计算公式

$$d_{L1}(x, y) = \sum_{i=1}^{d} |x_i - y_i|$$

  - 说明
    - 向量各维度绝对偏差的总和/标量绝对差值，衡量两个变量/向量差异程度

研究现状

- 歧视样本生成方法
  - 主流方法都采用“两阶段”生成方法，新的方法都是修改两阶段中的生成细节，易生成大量相似的歧视样本，费时费力且修复效果不好
- 对抗网络方法
  - 向DNN模型引入额外的对抗模块重新训练DNN模型来修复不公平问题，对抗网络的训练会导致更高的计算开销和复杂的超参数调优且无法了解其决策过程
- 模型修复方法
  - 通过对模型添加dropout层或者隐藏层进行再训练，过程较为繁琐，效率较低
  - 数据规模较小且没有考虑图像等非结构化数据

RUNNER: Responsible UNfair NEuron Repair for Enhancing Deep Neural Network Fairness

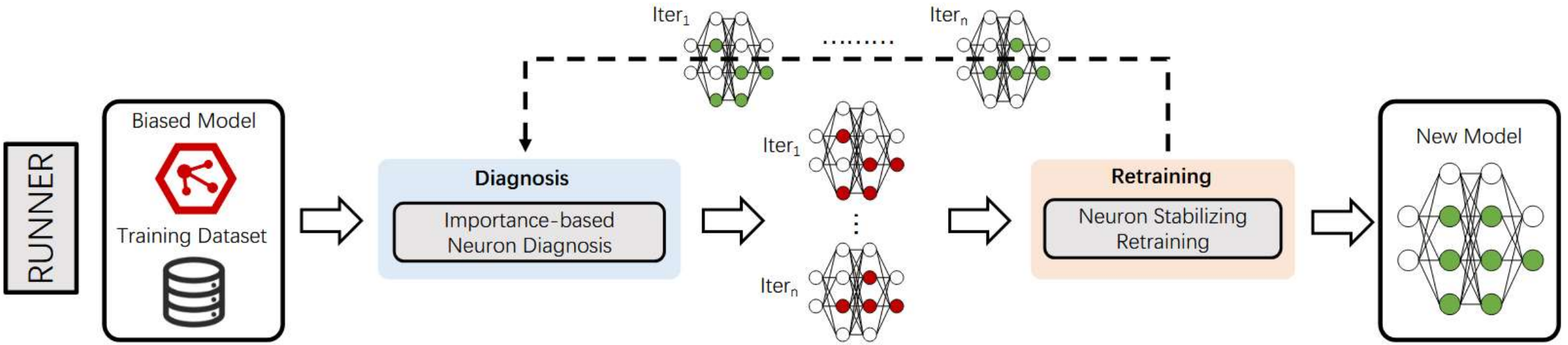
算法原理 TIPO

T	目标	保证准确率前提下提高公平性修复效率、效果和泛化性
I	输入	数据集*5，有偏模型*1
P	处理	1.基于重要性的神经元诊断 2.神经元稳定再训练，降低目标神经元的歧视 3.迭代修复策略，解决再训练后新不公平神经元出现的问题
O	输出	修复后提升公平性的新模型

P	问题	现有训练对抗网络的方法计算复杂度高，存在模式崩溃问题
C	条件	白盒模型能够计算梯度、能够访问训练数据
D	难点	1.如何在公平性和准确性之间找到平衡 2.如何适配非结构化数据
L	水平	ICSE 2024 (CCF-A)

RUNNER 算法原理图

- 算法原理图
  - 步骤1：基于重要性的神经元诊断识别出对目标模型产生非公平性预测有重大影响的神经元
  - 步骤2：神经元稳定再训练（降低目标神经元的歧视）
  - 步骤3：迭代修复（每轮迭代遵循“诊断-再训练”，持续更新模型参数）



创新分析 基于重要性的神经元诊断

- 定位高神经元不公平重要性（UI）神经元
  - 计算DP和EO的宽松度量（relaxed measures）

$$Gap_{DP} = |E_x{\sim}P_0(f(x) - E_x{\sim}P_1(f(x)|$$

- 其中$P_0 = P(x|a = 0)$和$P_1 = P(x|a = 1)$分别是$a = 0$和$a = 1$条件下x的分布，函数$E(-)$用于计算这些分布下的期望，$f(x)$是模型的输出，本质就是不同敏感属性群体输出均值差

$$Gap_{EO} = \sum_{y\in\{0,1\}} |E_x{\sim}P_0^y(f(x)) - E_x{\sim}P_1^y(f(x))|$$

- 其中$P_0^1 = P(x|a = 0,\ y = 1)$表示在$a = 0$和$y = 1$条件下x的分布,本质就是真实标签$y = 0$和$y = 1$条件下群体输出均值差
- 这两个通过关联ΔDP和ΔEO以及神经元参数ω从而实现可微，解决了传统ΔDP和ΔEO无法反向传播的问题

创新分析 基于重要性的神经元诊断

– 计算每个神经元的不公平重要性UI（移除该神经元 前后DP和EO宽松度量的平方差）

$$UI_m^I = (Gap_{DP/EO}f(W) - Gap_{DP/EO}f(W|\omega_m^l = 0))^2$$

- 为了计算每个神经元的UI需要逐个移除每个神经元，对于参数较大的模型计算复杂度高

- 一阶泰勒展开化简 $Gap_{DP/EO}f(W) \approx Gap_{DP/EO}f(W|\omega_m^l = 0) + \frac{\partial Gap_{DP/EO}}{\partial \omega_m^l}\omega_m^l$

$$UI_m^I = (\frac{\partial Gap_{DP/EO}}{\partial \omega_m^l}\omega_m^l)^2$$

– 根据计算出的神经元重要性确定最具责任的不公平神经元

$$\Omega_k^I = topk(\omega l, UI^l)$$

- 模型F的神经元不公平重要性$\Omega_k = \Omega_k^{l_0}, \Omega_k^{l_1}, \cdots, \Omega_k^{l_{k-1}}$
- 选择的负责任的不公平神经元数量由超参数k控制

创新分析 神经元稳定再训练与迭代修复

- 计算神经元歧视
  - 计算第l层第m个神经元在不同子组的激活距离作为神经元歧视

$$\delta(m, l, f, P_0, P_1) = ||E_{x\sim P_0} f_m^l(x) - E_{x\sim P_1} f_m^l(x)||_l$$

    - 其中$P_0 = P(x|a = 0)$代表x在a=0条件下的分布，函数$E(.)$用于计算分布下的期望
  - 计算针对DP/EO指标的再训练损失项
    - $L = L_{cls} + \lambda \sum_l^{l\in L} \sum_m^{m\in\Omega^l} \delta(m, l, f, P_0, P_1)$
    - $L = L_{cls} + \lambda \sum_l^{l\in L} \sum_m^{m\in\Omega^l} \sum_Y^{Y\in\{0,1\}} \delta(m, l, f, P_0^Y, P_1^Y)$
    - 其中L_{cls}表示交叉熵损失项,λ控制公平性正则化项权重，$l \in L$表示所有层的神经元，$m \in \Omega^l$表示所有诊断出的不公平神经元
- 迭代修复
  - 在每次修复迭代中都包含基于重要性的神经元诊断和稳定性再训练
  - 反向传播更新模型参数，减轻每个新识别出的不公平神经元的神经元歧视

实验流程 数据资源

- 模型资源
  - 表格数据集
    - 隐藏层大小200的MLP
    - Adam优化器
    - ΔDP批处理大小1000
    - ΔEO批处理大小2000
    - 学习率0.001
  - 图像数据集
    - AlexNet和ResNet-18
    - Adam优化器
    - ΔDP批处理大小64
    - ΔEO批处理大小128
    - 学习率0.0001

数据集	数据集用途	敏感属性	样本数量	特征总数
Adult	人口普查	性别	45222	14
COMPAS	再犯罪风险预测	种族	6167	10
Credit	评估个人信用	性别	600	20
LSAC	法学院招生	性别	20800	4
CelebA	预测面部属性	性别	202599	40

实验流程 对比方法和评价指标

- 对比方法
  - Vanilla（仅使用交叉熵损失）
  - Oversample（预处理：更加频繁从稀有样本子组采样）
  - Reweighing（预处理：重新分配训练数据集元组权重）
  - Adversarial（处理中：训练对抗网络）
  - Fairneuron（处理中：神经元切片、选择性训练）
  - Fairsmote（前处理：为数据点少子组生成新数据点）
  - ROC（后处理：利用单个或一组概率分类器低置信度区域）
- 评价指标
  - ΔDP（人口统计学均等）
  - ΔEO（均衡几率）
  - AP（平均精度）

实验设计 对比实验

- RQ1:RUNNER的有效性
- 实验结论
  - RUNNER在所有数据集上ΔDP和ΔEO都能取得最佳公平性表现
  - RUNNER在大部分数据集上相较于基准方法AP略有下降
  - RUNNER在credit数据集平均AP高于基准方法

	Metric	Vanilla	Oversample	Reweighing	Adversarial	FairNeuron	FairSmote	ROC	RUNNER
Adult	AP	0.781 ± 0.009	0.767 ± 0.011	0.720 ± 0.016	0.765 ± 0.015	0.787 ± 0.002	0.781 ± 0.009	0.646 ± 0.012	0.766 ± 0.011
	ACC	0.850 ± 0.005	0.825 ± 0.005	0.811 ± 0.010	0.823 ± 0.010	0.857 ± 0.001	0.852 ± 0.005	0.845 ± 0.005	0.838 ± 0.005
	ΔDP	0.170 ± 0.021	0.080 ± 0.013	0.102 ± 0.023	0.066 ± 0.011	0.171 ± 0.000	0.146 ± 0.021	0.082 ± 0.010	**0.048 ± 0.012**
COMPAS	AP	0.643 ± 0.004	0.634 ± 0.008	0.637 ± 0.003	0.640 ± 0.010	0.649 ± 0.001	0.632 ± 0.007	0.501 ± 0.012	0.623 ± 0.008
	ACC	0.655 ± 0.003	0.651 ± 0.007	0.643 ± 0.005	0.653 ± 0.010	0.658 ± 0.002	0.654 ± 0.005	0.636 ± 0.007	0.649 ± 0.006
	ΔDP	0.174 ± 0.009	0.140 ± 0.018	0.029 ± 0.017	0.125 ± 0.074	0.190 ± 0.008	0.166 ± 0.018	0.049 ± 0.012	**0.006 ± 0.004**
Credit	AP	0.797 ± 0.023	0.791 ± 0.013	0.811 ± 0.008	0.863 ± 0.003	0.846 ± 0.003	0.811 ± 0.019	0.792 ± 0.015	0.840 ± 0.009
	ACC	0.682 ± 0.036	0.655 ± 0.034	0.608 ± 0.030	0.687 ± 0.015	0.711 ± 0.006	0.660 ± 0.018	0.695 ± 0.019	0.654 ± 0.017
	ΔDP	0.134 ± 0.027	0.109 ± 0.037	0.139 ± 0.035	0.124 ± 0.052	0.156 ± 0.018	0.161 ± 0.049	0.150 ± 0.030	**0.056 ± 0.029**
LSAC	AP	0.924 ± 0.005	0.927 ± 0.004	0.930 ± 0.002	0.918 ± 0.007	0.927 ± 0.004	0.913 ± 0.008	0.879 ± 0.003	0.924 ± 0.003
	ACC	0.839 ± 0.007	0.842 ± 0.003	0.758 ± 0.016	0.836 ± 0.006	0.855 ± 0.008	0.823 ± 0.014	0.847 ± 0.001	0.840 ± 0.002
	ΔDP	0.007 ± 0.004	0.008 ± 0.006	0.008 ± 0.005	0.023 ± 0.211	0.005 ± 0.001	0.007 ± 0.004	0.004 ± 0.003	**0.004 ± 0.003**

实验设计 对比实验

- RQ1:RUNNER的有效性
- 实验结论
  - RUNNER在所有数据集上ΔDP和ΔEO都能取得最佳公平性表现
  - RUNNER在大部分数据集上相较于基准方法AP略有下降
  - RUNNER在credit数据集平均AP高于基准方法

	Metric	Vanilla	Oversample	Reweighing	Adversarial	FairNeuron	FairSmote	ROC	RUNNER
Adult	AP	0.779 ± 0.013	0.762 ± 0.010	0.762 ± 0.010	0.761 ± 0.014	0.786 ± 0.001	0.785 ± 0.010	0.600 ± 0.016	0.767 ± 0.012
	ACC	0.850 ± 0.004	0.819 ± 0.006	0.819 ± 0.006	0.749 ± 0.159	0.857 ± 0.001	0.852 ± 0.003	0.825 ± 0.007	0.813 ± 0.008
	ΔEO	0.096 ± 0.038	0.141 ± 0.024	0.141 ± 0.024	0.102 ± 0.047	0.138 ± 0.004	0.104 ± 0.038	0.145 ± 0.029	**0.082 ± 0.023**
COMPAS	AP	0.641 ± 0.006	0.645 ± 0.014	0.645 ± 0.014	0.643 ± 0.005	0.649 ± 0.001	0.635 ± 0.004	0.523 ± 0.012	0.637 ± 0.007
	ACC	0.653 ± 0.005	0.653 ± 0.007	0.653 ± 0.007	0.653 ± 0.006	0.658 ± 0.002	0.654 ± 0.006	0.636 ± 0.008	0.651 ± 0.004
	ΔEO	0.348 ± 0.045	0.156 ± 0.017	0.156 ± 0.017	0.057 ± 0.017	0.353 ± 0.020	0.318 ± 0.039	0.246 ± 0.040	**0.046 ± 0.015**
Credit	AP	0.788 ± 0.014	0.784 ± 0.009	0.784 ± 0.009	0.861 ± 0.005	0.843 ± 0.006	0.808 ± 0.023	0.764 ± 0.032	0.838 ± 0.017
	ACC	0.650 ± 0.019	0.639 ± 0.022	0.639 ± 0.022	0.612 ± 0.013	0.721 ± 0.006	0.634 ± 0.021	0.636 ± 0.030	0.660 ± 0.120
	ΔEO	0.343 ± 0.068	0.260 ± 0.099	0.260 ± 0.099	0.197 ± 0.037	0.442 ± 0.001	0.297 ± 0.106	0.250 ± 0.098	**0.179 ± 0.090**
LSAC	AP	0.925 ± 0.004	0.930 ± 0.001	0.930 ± 0.001	0.913 ± 0.006	0.926 ± 0.003	0.916 ± 0.009	0.864 ± 0.006	0.908 ± 0.004
	ACC	0.841 ± 0.003	0.762 ± 0.013	0.762 ± 0.013	0.695 ± 0.024	0.855 ± 0.008	0.823 ± 0.025	0.700 ± 0.024	0.671 ± 0.011
	ΔEO	0.024 ± 0.013	0.023 ± 0.009	0.023 ± 0.009	0.024 ± 0.016	0.037 ± 0.005	0.048 ± 0.011	0.029 ± 0.011	**0.018 ± 0.020**

- RQ2:RUNNER的效率
  - 在能够有效提升公平性的三种方法中，RUNNER效率最高
  - FairSmote方法在较大数据集上需要生成更多数据，效率较低
  - FairNeuron方法时间显著更长
  - 其他方法与基准方法训练时间接近

	Adversarial		FairNeuron				FairSmote		RUNNER	
	epochs	time	epochs	time_{D}	time_{R}	time	epochs	time	epochs	time
Adult	15	83.4s	10	692.2s	36.7s	728.9s	5	100.9s	5	**66.5s**
COMPAS	15	22.7s	10	201.9s	15.5s	217.4s	5	**15.5s**	**5**	19.4s
Credit	15	21.4s	10	106.4s	10.7s	117.1s	5	10.1s	5	**9.8s**
LSAC	15	35.5s	10	345.1s	25.4s	370.5s	5	26.4s	5	**22.9s**

- RQ3:在图像领域的泛化性
  - RUNNER方法在ΔDP和ΔEO指标上始终能取得最佳公平性表现，特别是在“卷发”分类预测任务中
  - 过采样方法可以改善公平性指标，但对ΔDP影响较小
  - 对抗性方法也可持续提升公平性，但AP性能不稳定

Table 5: Results of "wavy hair" classification.

Metric	Vanilla	Oversample	Adversarial	FairNeuron	RUNNER
AP	0.829 ± 0.011	0.801 ± 0.039	0.806 ± 0.051	0.817 ± 0.015	0.805 ± 0.050
ACC	0.783 ± 0.019	0.780 ± 0.036	0.771 ± 0.039	0.775 ± 0.026	0.771 ± 0.049
ΔDP	0.250 ± 0.036	0.253 ± 0.064	0.241 ± 0.054	0.245 ± 0.049	**0.239 ± 0.086**
Metric	Vanilla	Oversample	Adversarial	FairNeuron	RUNNER
AP	0.724 ± 0.044	0.774 ± 0.025	0.748 ± 0.041	0.766 ± 0.071	0.776 ± 0.013
ACC	0.701 ± 0.025	0.747 ± 0.061	0.715 ± 0.072	0.739 ± 0.053	0.733 ± 0.088
ΔEO	0.219 ± 0.070	0.089 ± 0.067	0.097 ± 0.054	0.269 ± 0.099	**0.078 ± 0.040**

- RQ3:在图像领域的泛化性
  - RUNNER方法在ΔDP和ΔEO指标上始终能取得最佳公平性表现，特别是在“卷发”分类预测任务中
  - 过采样方法可以改善公平性指标，但对ΔDP影响较小
  - 对抗性方法也可持续提升公平性，但AP性能不稳定

Table 4: Results of "attractive" classification.

Metric	Vanilla	Oversample	Adversarial	FairNeuron	RUNNER
AP	0.881 ± 0.024	0.894 ± 0.015	0.859 ± 0.033	0.880 ± 0.024	0.866 ± 0.008
ACC	0.780 ± 0.037	0.800 ± 0.017	0.756 ± 0.040	0.783 ± 0.035	0.766 ± 0.009
ΔDP	0.450 ± 0.019	0.451 ± 0.018	0.282 ± 0.021	0.448 ± 0.018	**0.215 ± 0.016**

Metric	Vanilla	Oversample	Adversarial	FairNeuron	RUNNER
AP	0.821 ± 0.031	0.864 ± 0.003	0.813 ± 0.022	0.837 ± 0.041	0.867 ± 0.006
ACC	0.718 ± 0.034	0.769 ± 0.010	0.711 ± 0.024	0.734 ± 0.036	0.781 ± 0.005
ΔEO	0.496 ± 0.055	0.056 ± 0.011	0.134 ± 0.035	0.513 ± 0.036	**0.050 ± 0.013**

实验流程 实验结果

- RQ4:可配置超参数的影响
  - 在多个K的超参数设置下，RUNNER始终能够有效地修复公平性
  - 在不同场景下K的最优值并不统一
  - 在大多数超参数配置下RUNNER可以取得比其他基线方法更加优异的公平性表现

Table 6: Comparison of various hyperparameter k settings on the Adult Dataset.

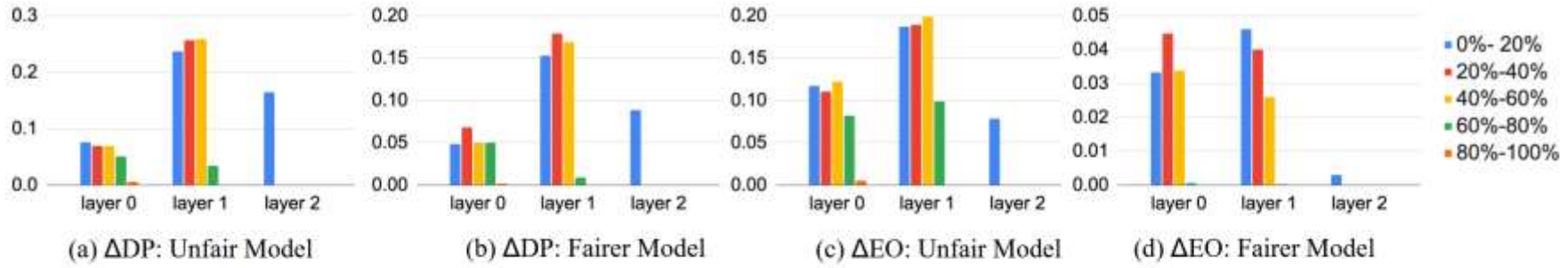
Adult	5%	10%	20%	50%
AP	0.766 ± 0.013	0.766 ± 0.011	0.762 ± 0.008	0.751 ± 0.013
ACC	0.839 ± 0.006	0.838 ± 0.005	0.836 ± 0.005	0.833 ± 0.005
ΔDP	0.053 ± 0.011	0.048 ± 0.012	0.042 ± 0.012	**0.023 ± 0.014**
	5%	10%	20%	50%
AP	0.762 ± 0.013	0.767 ± 0.012	0.763 ± 0.014	0.762 ± 0.009
ACC	0.810 ± 0.006	0.813 ± 0.008	0.814 ± 0.004	0.807 ± 0.005
ΔEO	**0.073 ± 0.026**	0.082 ± 0.023	0.085 ± 0.026	0.082 ± 0.027

Table 7: Comparison of various hyperparameter k settings on the CelebA Dataset for "attractive" classification.

CelebA	5%	10%	20%	50%
AP	0.858 ± 0.006	0.861 ± 0.007	0.860 ± 0.004	0.866 ± 0.008
ACC	0.751 ± 0.019	0.761 ± 0.019	0.758 ± 0.016	0.766 ± 0.009
ΔDP	**0.189 ± 0.033**	0.197 ± 0.020	0.202 ± 0.029	0.215 ± 0.016
	5%	10%	20%	50%
AP	0.865 ± 0.007	0.861 ± 0.012	0.867 ± 0.006	0.863 ± 0.009
ACC	0.775 ± 0.012	0.773 ± 0.011	0.781 ± 0.005	0.774 ± 0.010
ΔEO	0.068 ± 0.023	0.053 ± 0.013	**0.050 ± 0.013**	0.064 ± 0.025

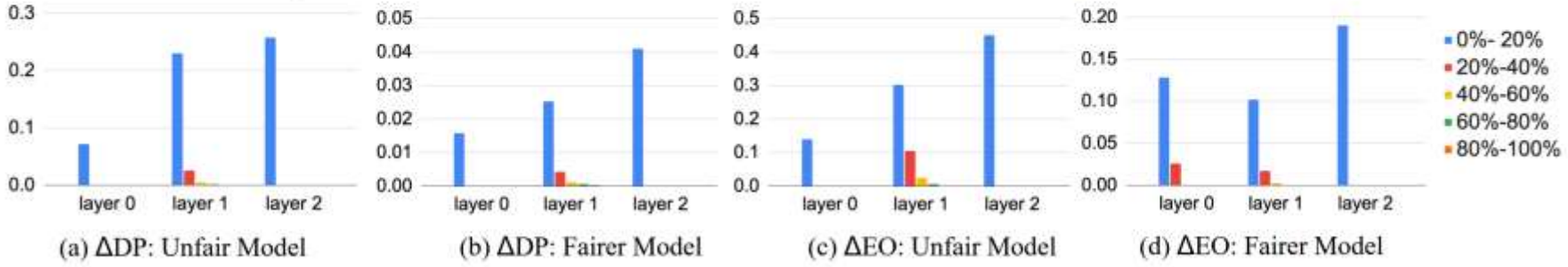
实验流程 实验结果–不公平程度

- RQ5:不公平重要性分数UI与神经元歧视分数的相关性
  - 在两个数据集两种公平性指标下不公平模型歧视值都高于公平模型
  - Adult数据集前20%神经元与20%-40%歧视程度接近，高于后面神经元
  - CelebA数据集上歧视主要存在于前20%的神经元中



(a) ΔDP: Unfair Model (b) ΔDP: Fairer Model (c) ΔEO: Unfair Model (d) ΔEO: Fairer Model

Figure 3: Neuron discrimination on the Adult Dataset on the ΔDP and ΔEO metric.

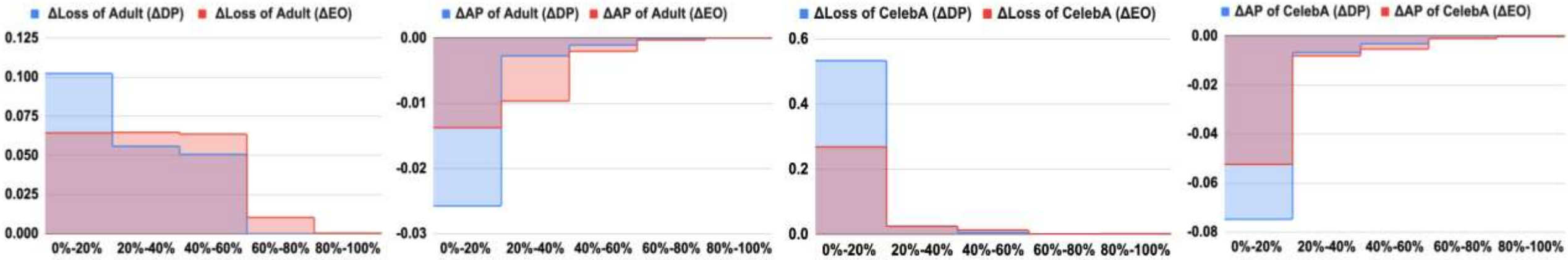


(a) ΔDP: Unfair Model (b) ΔDP: Fairer Model (c) ΔEO: Unfair Model (d) ΔEO: Fairer Model

Figure 4: Neuron discrimination on the CelebA Dataset on the ΔDP and ΔEO metric.

实验流程 实验结果-负责任程度

- RQ6:神经元对最终模型预测的重要性
  - 随着UI值的降低，ΔAP和Δloss更接近于0
  - CelebA数据集上对最后80%的神经元进行dropout几乎不会影响损失和AP值
  - 高UI值的神经元对最终预测的影响更大



(a) Δloss of Adult (b) ΔAP of Adult (c) Δloss of CelebA (d) ΔAP of CelebA

Figure 5: Dropout results on the Adult and CelebA Dataset under the ΔDP and ΔEO metric.

特点总结与未来展望

算法总结 RUNNER

- 算法流程
  - 基于重要性的神经元诊断
  - 神经元稳定再训练
  - 迭代修复
- 算法优势
  - 公平性修复效果好，效率高
  - 无需生成额外的歧视样本和训练对抗网络
  - 在图像数据集上泛化性强
- 未来展望
  - 探索不同的距离度量来更加准确量化神经元歧视，改进诊断过程
  - 使用此方法修复DNN模型的其他缺陷

参考资源

[1]. Tianlin Li, Yue Cao, Jian Zhang, Shiqian Zhao, Yihao Huang, Aishan Liu, Qing Guo, and Yang Liu. RUNNER: Responsible UNfair NEuron Repair for Enhancing Deep Neural Network Fairness, IEEE/ACM 46th International Conference on Software Engineering (ICSE '24) [C]. New York, NY, USA, Association for Computing Machinery, 2024, Article 9, 1–13.

道可道，非常道。名可名，非常名。无名天地之始。有名万物之母。故常无欲以观其妙。常有欲以观其徼。此两者同出而异名，同谓之玄。玄之又玄，众妙之门。

谢谢！