软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2024,35(6):2936−2950 [doi: 10.13328/j.cnki.jos.006928]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

谛听: 面向鲁棒分布外样本检测的半监督对抗训练方法*

周志阳[1,2], 窦文生[1,2,3], 李 硕[1,2], 亢良伊[1,2], 王 帅[1,2], 刘 杰[1,2,3,4], 叶 丹[1,2,3]

[1](中国科学院 软件研究所, 北京 100190)

[2](中国科学院大学, 北京 100049)

[3](计算机科学国家重点实验室 (中国科学院 软件研究所), 北京 100190)

[4](中国科学院大学南京学院, 江苏 南京 211135)

通信作者: 刘杰, E-mail: ljie@otcaix.iscas.ac.cn; 叶丹, E-mail: yedan@otcaix.iscas.ac.cn

摘 要: 检测训练集分布之外的分布外 (out-of-distribution, OOD) 样本对于深度神经网络 (deep neural network, DNN) 分类器在开放环境的部署至关重要. 检测 OOD 样本可以视为一种二分类问题, 即把输入样本分类为“分布内 (in-distribution, ID)”类或“分布外”类. 进一步地, 检测器自身还可能遭受到恶意的对抗攻击而被再次绕过. 这些带有恶意扰动的 OOD 样本称为对抗 OOD 样本. 构建鲁棒的 OOD 检测器以检测对抗 OOD 样本是一项更具挑战性的任务. 为习得可分离且对恶意扰动鲁棒的表示, 现有方法往往利用辅助的干净 OOD 样本邻域内的对抗 OOD 样本来训练 DNN. 然而, 由于辅助的 OOD 训练集与原 ID 训练集的分布差异, 训练对抗 OOD 样本无法足够有效地使分布内决策边界对对抗扰动真正鲁棒. 从 ID 样本的邻域内生成的对抗 ID 样本拥有与原 ID 样本近乎一样的语义信息, 是一种离分布内区域更近的 OOD 样本, 对提升分布内边界对对抗扰动的鲁棒性很有效. 基于此, 提出一种半监督的对抗训练方法——谛听, 来构建鲁棒的 OOD 检测器, 用以同时检测干净 OOD 样本和对抗 OOD 样本. 谛听将对抗 ID 样本视为一种辅助的“近 OOD”样本, 并将其与其他辅助的干净 OOD 样本和对抗 OOD 样本联合训练 DNN, 以提升 OOD 检测的鲁棒性. 实验结果表明, 谛听在检测由强攻击生成的对抗 OOD 样本上具有显著的优势, 同时在原分类主任务及检测干净 OOD 样本上保持先进的性能.

关键词: 分布外样本检测; 对抗鲁棒性; 对抗训练

中图法分类号: TP18

中文引用格式: 周志阳, 窦文生, 李硕, 亢良伊, 王帅, 刘杰, 叶丹. 谛听: 面向鲁棒分布外样本检测的半监督对抗训练方法. 软件学报, 2024, 35(6): 2936–2950. http://www.jos.org.cn/1000-9825/6928.htm

英文引用格式: Zhou ZY, Dou WS, Li S, Kang LY, Wang S, Liu J, Ye D. DiTing: Semi-supervised Adversarial Training Approach for Robust Out-of-distribution Detection. Ruan Jian Xue Bao/Journal of Software, 2024, 35(6): 2936–2950 (in Chinese). http://www.jos.org.cn/1000-9825/6928.htm

DiTing: Semi-supervised Adversarial Training Approach for Robust Out-of-distribution Detection

ZHOU Zhi-Yang[1,2], DOU Wen-Sheng[1,2,3], LI Shuo[1,2], KANG Liang-Yi[1,2], WANG Shuai[1,2], LIU Jie[1,2,3,4], YE Dan[1,2,3]

[1](Institute of Software, Chinese Academy of Sciences, Beijing 100190, China)

[2](University of Chinese Academy of Sciences, Beijing 100049, China)

[3](State Key Laboratory of Computer Science (Institute of Software, Chinese Academy of Sciences), Beijing 100190, China)

[4](University of Chinese Academy of Sciences, Nanjing, Nanjing 211135, China)

Abstract: Detecting out-of-distribution (OOD) samples outside the training set distribution is crucial for deploying deep neural network (DNN) classifiers in the open environment. OOD sample detection is a binary classification problem, which is to classify the input samples

* 基金项目: 国家自然科学基金 (61972386)
收稿时间: 2022-10-19; 修改时间: 2023-01-15; 采用时间: 2023-03-02; jos 在线出版时间: 2023-09-13
CNKI 网络首发时间: 2023-09-15

into the in-distribution (ID) or OOD categories. Then, the detector itself can be re-bypassed by malicious adversarial attacks. These OOD samples with malicious perturbations are called adversarial OOD samples. Building robust OOD detectors to detect adversarial OOD samples is more challenging. Existing methods usually train DNN through adversarial OOD samples within the neighborhood of auxiliary clean OOD samples to learn separable and robust representations to malicious perturbations. However, due to the distributional differences between the auxiliary OOD training set and original ID training set, training adversarial OOD samples is not effective enough to ensure the robustness of ID boundary against adversarial perturbations. Adversarial ID samples generated from within the neighborhood of (clean) ID samples are closer to the ID boundary and are also effective in improving the adversarial robustness of the ID boundary. This study proposes a semi-supervised adversarial training approach, DiTing, to build robust OOD detectors to detect clean and adversarial OOD samples. This approach treats the adversarial ID samples as auxiliary "near OOD" samples and trains them jointly with other auxiliary clean and adversarial OOD samples to improve the robustness of OOD detection. Experiments show that DiTing has a significant advantage in detecting adversarial OOD samples generated by strong attacks while maintaining state-of-the-art performance in classifying clean ID samples and detecting clean OOD samples.

Key words: out-of-distribution sample detection; adversarial robustness; adversarial training

深度神经网络 (deep neural network, DNN) 在图像识别[1]、自动驾驶[2]和医学诊断[3]等各个领域都取得了前所未有的性能, 广泛地应用于各种对安全敏感的领域. 然而, DNN 分类器容易对来自训练集分布之外的分布外 (out-of-distribution, OOD) 样本过信心[4], 以较高的预测信心而产生误分类. 例如, 将一张"键盘"的图片输入给一个在"猫"和"狗"数据集上训练的 DNN 分类器, 其可能以 90% 以上的 Softmax 信心将该图片分类为"猫". 检测 OOD 样本对 DNN 在开放环境的部署至关重要.

检测 OOD 样本是一种二分类问题, 即把输入样本分类为"分布内"类或"分布外"类. 当前多数检测方法可以划分为两种方法路线. 第 1 种路线侧重于为已有 (pre-trained) DNN 模型设计基于阈值的检测/打分函数 (scoring function) 来检测 OOD 样本[5−8]. 当输入样本的分数小 (大) 于某阈值时, 则将其判断为 OOD 样本. 在这些打分函数中, 较高效或有效的且与我们的工作相关的是基于统计的打分函数, 如基于 DNN 的最大 Softmax 概率 (maximum Softmax probability, MSP)[5,6,9]的打分函数. 基于统计的打分函数将 DNN 视为一个特征提取器, 并使用其隐藏层或/和输出层所蕴含的信息作为输入来区分 ID 样本和 OOD 样本.

第 2 种方法侧重于重新训练 DNN, 以习得对 ID 样本和 OOD 样本可分离的表示. Lee 等人[9]发现使用围绕在分布内区域附近的 OOD 样本对压缩分布内区域更有效; 他们同时训练分类器和 GAN, 使分类器在 GAN[10]生成的"边界"数据上输出均匀分布来帮助校验该分类器的预测信心 (即使 DNN 对测试样本的平均预测信心接近于其准确率). 半监督的 OE[6]进一步地使用多样的、真实世界的辅助 OOD 数据集来帮助校验 DNN 的预测信心; OE 训练 DNN 对辅助的 OOD 样本输出一个均匀分布, 并首次大幅度地提升了检测 OOD 样本的性能. 紧随 OE 之后, 半监督的 SSL[11]使用多个额外的"拒绝"类来表示 OOD 样本, 并相较于 OE 取得了进一步的性能提升. 总结来看, 检测 OOD 样本问题得到了较好的解决[12,13].

然而, 与 DNN 分类器的弱鲁棒性类似, 最近的工作[14−16]表明多数先进的 OOD 检测方法同样对恶意的对抗扰动[17−19]敏感,易被注入了对抗扰动的 OOD 样本再次绕过. 例如, 攻击者可通过注入一些对抗扰动到分布外的广告牌上, 以骗过自动驾驶系统将并其识别为"右转"标识. 为了后续便于描述, 本文把无恶意目的 OOD 样本称为干净 OOD 样本, 把为了绕过 OOD 检测器而注入了恶意扰动的 OOD 样本称为对抗 OOD 样本. 检测对抗 OOD 样本是一项更具挑战性的任务. 受对抗训练 (adversarial training, AT)[20,21]的启发, 已有工作大多在辅助的 OOD 样本上直接引入 AT 来帮助提升 OOD 检测器的鲁棒性. 在常规鲁棒性研究领域, AT 为对抗样本分配与其干净样本一样的标签, 将对抗样本视为一种数据增强来训练 DNN 分类器. 尽管 AT 有效地保证了 DNN 的鲁棒性, 但其强制 DNN 完全忽略与标签弱相关的扰动特征来辅助决策, 导致 DNN 在原 ID 样本上的分类准确率显著下降[22]. Hein 等人[15]分析了为什么使用 ReLU 激活函数的 DNN 易对 OOD 样本产生高置信度, 并提出了 ACET 在辅助的 OOD 样本上引入 AT 来帮助缓解此问题; ACET 训练 DNN 对干净 OOD 样本和对抗 OOD 样本一个均匀分布的预测概率. ATOM[16]进一步提出一种辅助 OOD 样本挖掘策略, 并使用第 K+1"拒绝"类来专有地表示干净 OOD 样本和对抗 OOD 样本. 此外, ALOE[12]和 RATIO[13]在 ID 样本和辅助的 OOD 样本上同时引入常规的 AT 而不使用任何干净

的 ID 样本和干净 OOD 样本来训练 DNN.

这些方法都声称在辅助 OOD 样本上应用 AT 提升了 OOD 检测的鲁棒性. 然而, 由于辅助的 OOD 训练集与原 ID 训练集的分布差异, 本文发现训练辅助的干净 OOD 样本邻域内的对抗 OOD 样本不能有效地使分布内决策边界对更强的攻击鲁棒 (详见第 2 节的实证); 在 ID 样本上直接应用常规的 AT 会导致原主任务性能 (即在干净 ID 样本上的分类准确率) 显著降低, 同样是一种次优的解决方案. 从 ID 样本邻域内生成的对抗 ID 样本拥有与原干净 ID 样本近乎一致的语义信息 (例如, 一张被注入对抗扰动的“猫”的照片在视觉上与其原干净照片近乎一样), 离分布内区域更近, 是一种“近 OOD”样本. 与对抗样本检测[5,7,8,20,23−27]研究领域类似, 本文把对抗 ID 样本视为一种辅助的“近 OOD”样本, 提出一种半监督对抗训练方法——谛听, 来构建鲁棒的 OOD 检测器, 用以同时检测干净 OOD 样本和对抗 OOD 样本. 谛听不仅使用辅助的干净 OOD 样本和对抗 OOD 样本, 也同时使用辅助的对抗 ID 样本来联合训练 DNN.

图 1 展示了谛听的直观示例图, 其中, 最后一层中的虚线节点是在 DNN 的最后一层添加的用于表示分布外样本 (对抗 ID 样本、干净 OOD 样本和对抗 OOD 样本) 的拒绝类. 图 1 中假设原训练集真实类别的数量为 2, 额外拒绝类的数量也为 2. 在步骤 1 中, 我们固定住 DNN 的权重参数 θ, 使用 (干净) ID 样本 x_1^{in} 和辅助的干净 OOD 样本 x_1^{o} 来分别创建对抗 ID 样本 $x_1^{\text{in}}+\delta_1^{\text{in}}$ 和对抗 OOD 样本 $x_1^{\text{o}}+\delta_1^{\text{o}}$ (详见第 3.2 节). 在步骤 2 中, 我们为对抗 ID 样本 $x_1^{\text{in}}+\delta_1^{\text{in}}$ 设置了与其原 ID 样本 x_1^{in} 不同的伪标签 $[0,0,1,0]^{\text{T}}$ (详见第 3.1 节), 并将 $x_1^{\text{in}}+\delta_1^{\text{in}}$ 和 x_1^{in} 成对地喂入 DNN, 以使 DNN 能够更好地学习它们之间的差异 δ_1^{in}; 与此同时, 干净 OOD 样本 x_1^{o} 及其对抗 OOD 样本 $x_1^{\text{o}}+\delta_1^{\text{o}}$ 都标注了伪标签 $[0,0,1,0]^{\text{T}}$, 这使 DNN 在 OOD 样本上习得对对抗扰动 δ_1^{o} 的不变性以及对 ID 样本的差异. 需要注意的是, 在常规 AT 中, $x_1^{\text{in}}+\delta_1^{\text{in}}$ 分配了与 x_1^{in} 相同的标签 $[1,0,0,0]^{\text{T}}$; 而在谛听中, $x_1^{\text{in}}+\delta_1^{\text{in}}$ 的伪标签是 $[0,0,1,0]^{\text{T}}$ (即位于拒绝类上), 这使得谛听并不强迫 DNN 忽略那些“类似于” δ_1^{in} 的细微特征, 从而不会引起 DNN 在原分类任务上的干净准确率显著下降.

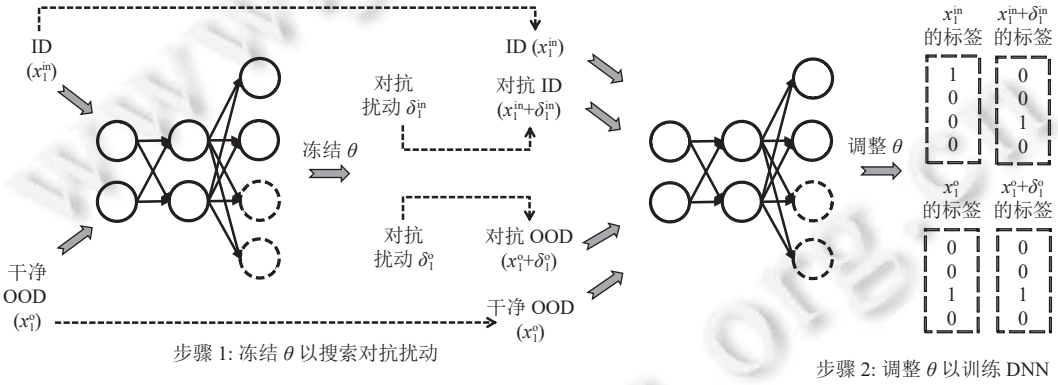

图 1 谛听的直观示例图

为了可靠地评估 OOD 检测器的鲁棒性以避免过高评估, 本文基于先进的 Auto-PGD[28]搜索算法开发更强的攻击来评估 OOD 检测器的鲁棒性. 实验表明, 谛听在检测由强攻击 (即 Auto-PGD 系列的攻击) 所产生的对抗 OOD 样本上比已有方法具备显著的性能优势, 同时在原分类干净 ID 样本和检测干净 OOD 样本上保持先进的性能. 本文工作开源地址: https://gitee.com/zhiyang3344/diting. 总体而言, 本文贡献如下.

(1) 本文首次揭示, 训练辅助的对抗 OOD 样本无法足够有效地使得分布内边界对对抗扰动鲁棒, 训练对抗 ID 样本作为辅助的 OOD 样本能有效地提升 OOD 检测器的鲁棒性.

(2) 本文提出一种用于提升 OOD 检测器鲁棒性的半监督对抗训练方法——谛听, 其不仅使用辅助的干净 OOD 样本和对抗 OOD 样本, 也使用辅助的对抗 ID 样本作为 OOD 样本来联合地训练 DNN, 用以提升 OOD 检测的鲁棒性. 不同于常规的对抗训练, 谛听并不会显著地损害 DNN 分类器在原分类任务上的性能.

(3) 基于先进的 Auto-PGD 算法, 本文开发并开源了一系列用于评估 OOD 检测器鲁棒性的、攻击力更强的攻

击算法.

(4) 实验结果表明, 在检测由更强的攻击生成的对抗 OOD 样本上, 谛听比已有方法具备显著的性能优势, 同时在分类干净 ID 样本和检测干净 OOD 样本上保持先进的性能.

本文第 1 节介绍对抗攻击、对抗训练和分布外样本检测等相关工作. 第 2 节实证仅训练辅助的对抗 OOD 样本的不足以及训练对抗 ID 样本对提升 OOD 检测器鲁棒性的影响. 第 3 节对本文所提的半监督对抗训练方法——谛听及其实现进行介绍. 第 4 节通过实验验证了所提谛听在保证 OOD 检测器鲁棒性方面的有效性. 最后对本文进行总结与展望.

1 相关工作

1.1 对抗攻击

当 DNN 在各个领域都取得了前所未有的性能, 已有工作[17]表明 DNN 容易受到对抗扰动的影响. 随后, 研究人员提出了各种对抗性攻击方法[17−19,29−35]. FGSM (fast gradient sign method)[18]使用有关于输入的损失梯度的符号值 (即 $sign(\cdot)$) 来高效地制作对抗样本. R+FGSM[36]引入了一个随机扰动步骤到 FGSM 以增加攻击成功率. 多步迭代的 BIM[20,32,37]进一步考虑使用多步梯度迭代. 这些攻击可以归类为术语为 K-步的 projected gradient descent (PGD-K).

$$\delta^{k+1}=Proj_{\|x'-x\|_p\leqslant\epsilon}\left(\delta^k+\alpha\cdot sign\left(\nabla_{\delta^k}\ell\left(f_\theta\left(x+\delta^k\right),y\right)\right)\right) \tag{1}$$

其中, δ^k 表示第 k 步的扰动, $\ell(f_\theta(x+\delta^k),y)$ 表示受害模型 f_θ 在输入 $x+\delta^k$ 及其标签 y 上的对抗损失 (例如, 交叉熵损失 ℓ_{CE}), ∇_{δ^k} 表示关于 δ^k 的梯度, α 表示攻击步长, $\|\cdot\|_p$ 表示 L_p Norm 约束, $Proj_{\|x'-x\|_p\leqslant\epsilon}$ 表示投影搜索到的对抗样本 x' 到干净样本 x 的 ϵ -ball 上. 初始的随机扰动 δ^0 满足 $\left\|\delta^0\right\|_p\leqslant\epsilon$. 此外, 如果把公式 (1) 中的 y 换成其他非正确类别的标签并反转损失 ℓ 的符号, 则公式 (1) 将变成有目标的攻击. 为了识别虚假的防御[38]如蒸馏防御[39], CW[40]攻击直接攻击 DNN 的 logits 层输出 (即最后一层施加 Softmax 激活函数之前).

$$\ell_{\mathrm{CW}}=-(z_\theta(x+\delta)_y-\max_{i\neq y}(z_\theta(x+\delta)_i)) \tag{2}$$

其中, $z_\theta(x+\delta)_y$ 表示与标签 y 对应的 logit, $\max_{i\neq y}(z_\theta(x+\delta)_i)$ 表示除 $z_\theta(x+\delta)_y$ 外最大 logit. 为了进一步提升攻击能力, 文献 [41] 提出了多目标 (multi-targeted) 的 PGD 攻击, 其轮流使用其他非正确的类作为攻击目标来执行攻击. 最近的文献 [28] 提出了 Auto-PGD, 其集成了“动量更新”“攻击重启动”和“攻击步长自动调整”到 PGD 中; Auto-PGD 攻击可以更有效地使搜索避免陷入局部最优点, 在鲁棒性研究领域被广泛地应用于评估 DNN 分类器的鲁棒性. 此外, 检测-感知的自适应攻击 (adaptive attack) 及其变种[27,36]广泛地用于检测对抗 (ID) 样本的评估. 在本文中, 我们结合自适应攻击和 Auto-PGD 开发更强的攻击来评估 OOD 检测器的鲁棒性, 以避免对 OOD 检测器的过高估计.

1.2 对抗训练

为了防御对抗攻击, 研究者们提出了多种防御方法[18,20,23,24,29,39]. 然而, 此中大多数的防御被证明都只是呈现了如梯度混淆[38]的虚假安全, 并被后来更强的攻击击败, 几乎只有经验性的对抗训练 (adversarial training, AT)[18,20]可以有效地保证 DNN 的真正鲁棒性[38,42]. AT 将对抗样本视为干净 ID 样本的一种数据增强来训练 DNN. AT 的 min-max 框架如下.

$$\arg\min_\theta\frac{1}{N}\sum_{i=1}^N\max_{\|\delta\|_p\leqslant\epsilon}\ell(f_\theta(x_i+\delta_i),y_i) \tag{3}$$

其中, N 是训练样本的数量, x_i 表示第 i 个干净样本, ℓ 一般指交叉熵损失. 在公式 (3) 的内部 max 中, PGD-K 攻击常用于近似地搜索最优扰动 δ_i^* . AT 使 DNN 习得对抗性扰动的不变性. 然而, 正如文献 [22] 所指出的, AT 强制 DNN 忽略那些与标签弱相关的特征 (即那些易受扰动干扰的、难以察觉的特征) 来进行预测, 导致 DNN 在干净样本上的准确率显著下降.

1.3 分布外样本检测

随着 DNN 可靠性受到越来越多的关注, 研究者们提出了大量的方法来检测 OOD 样本的[4]. 检测 OOD 样本

是一个二分类问题, 即把输入样本分类为“分布内”类或“分布外”类. 本文将这些方法分为两大类. 第 1 类侧重于设计一些基于阈值的打分函数. 其中, 基于统计的打分函数是比较高效或有效的且与本文最相关的. 基于统计的打分函数[7,8,24,25]通常利用 DNN 的隐层或/和输出层所蕴含的信息作为输入来判断样本是否是 OOD 样本. 常见的基于统计的打分函数如基于最大 Softmax 概率 (maximum Softmax probability, MSP)[5,9]和基于马氏距离[8]的打分函数等. 第 2 类方法侧重于重新训练 DNN 以习得更加可分离的表示. Lee 等人[9]发现使用围绕在分布内区域附近的辅助 OOD 样本对压缩分布内区域更有效; 他们同时训练分类器和 GAN[10], 使分类器在 GAN 生成的“边界”数据上输出均匀分布帮助该校验该分类器的预测信心. 半监督的 OE[6]使用大量的、真实世界的辅助 OOD 数据集来训练 DNN; OE 强迫 DNN 在辅助 OOD 数据上数据较低置信度来帮助提升其不确定性估计, 并首次大幅度地提升了检测 OOD 样本的性能. 紧随 OE, 半监督的 SSL[36]使用额外的多个“拒绝”类别来专门表示 OOD 样本, 并取得了更优的性能. 总的来讲, 检测 (干净) OOD 样本已得到了较多的研究且得到了较好解决.

然而, 由于 DNN 自身的脆弱性, 已有工作[14,15,36]发现多数先进的 OOD 检测方法对对抗扰动敏感, 易被注入了恶意扰动的对抗 OOD 样本再次绕过. 为了方便后续描述, 本文将有无注入恶意攻击的 OOD 样本分别称为对抗 OOD 样本和干净 OOD 样本. ACET[10]分析了为什么使用 ReLU 激活函数的 DNN 易对远离分布内的 OOD 样本产生高置信度, 并在辅助的 OOD 样本上引入对抗训练 (AT) 来帮助缓解此问题. ATOM[36]根据干净 OOD 样本在 DNN 上的置信度, 进一步提出了一种 OOD 样本挖掘策略以提升 OOD 检测的鲁棒性. ACET 和 ATOM 的训练目标可以统一表示如下:

$$\underset{\theta}{\operatorname{argmin}}\frac{1}{N}\sum_{i=1}^{N}\ell(f_\theta(x_i^{\text{in}}),y_i^{\text{in}})+\beta\cdot\frac{1}{M}\left[\sum_{j=1}^{M_C}\ell(f_\theta(x_j^{\text{o}}),y_j^{\text{o}})+\sum_{k=M_C+1}^{M}\ell(f_\theta(x_k^{\text{o}}+\delta_k^{\text{o}*}),y_k^{\text{o}})\right] \tag{4}$$

其中, N 表示 (干净) ID 样本的总数, x_i^{in} 表示标签为 y_i^{in} 的第 i 个 ID 样本 (对应于公式 (3) 的 x_i), M 表示辅助 OOD 样本的总数量, M_C 表示干净样本的数量 ($M-M_C$ 表示对抗 OOD 样本的数量), x_j^{o} 表示第 j 个干净 OOD 样本, y_j^{o} 表示 x_j^{o} 的伪标签, $x_k^{\text{o}}+\delta_k^{\text{o}*}$ 表示从干净 OOD 样本 x_k^{o} 的 ϵ-ball 邻域内搜索到的对抗 OOD 样本. 在 ACET 中, y^{o} 是一个 K 维的均匀分布 (假设 ID 训练集包含 K 种类别); 而在 ATOM 中, y^{o} 则表示第 K+1 拒绝类. 与常规 AT 一样, 公式 (5) 使用 PGD-K 来最大化训练数据及其标签上的负数据似然以近似求解最优扰动 $\delta_k^{\text{o}*}=\operatorname{argmax}_{\|\delta_k^{\text{o}}\|\leqslant\epsilon}\ell(f_\theta(x_k^{\text{o}}+\delta_k^{\text{o}}),y_k^{\text{o}})$. 虽然 ACET 和 ATOM 都在 OOD 样本上引入了 AT, 但由于 ID 样本和干净 OOD 样本的分布差异, 训练干净 OOD 样本邻域内的对抗 OOD 样本无法有效地使分布内边界对对抗扰动足够鲁棒. 此外, ALOE[12]和 RATIO[13]在 ID 样本和辅助的干净 OOD 样本上都引入公式 (3) 中的常规 AT 以尝试提升 OOD 检测器的鲁棒性. 然而, 常规的 AT 将导致 DNN 在原主任务性能 (即在干净 ID 样本上的分类准确率) 的显著降低, 同样是一种次优的解决方案. 本文将 ALOE 和 RATIO 的方法标记为 $\text{AT}_{\text{out}}^{\text{in}}$, 并在第 4.2 节对该类方法展开进一步地实验对比和分析.

2 研究动机

本文首先实证研究训练辅助的对抗 OOD 样本能否有效地使分布内决策边界对对抗扰动真正鲁棒; 然后, 本文验证训练对抗 ID 样本作为辅助的 OOD 样本对分布内边界鲁棒性的影响.

2.1 训练辅助的对抗 OOD 样本

在常规的训练鲁棒的 DNN 分类器的任务中, 常规的对抗训练 (AT) 训练由 PGD 攻击生成的对抗扰动可以良好地泛化到其他攻击生成的扰动上, 从而使 DNN 不同类别间的分类边界变得对对抗扰动不敏感. 常规 AT 有效保证了 DNN 对 ID 样本邻域内的对抗扰动的鲁棒性, 但是无法保证在 (与 ID 样本具有语义差别的) OOD 样本上的鲁棒性. 受常规 AT 的启发, 在构建鲁棒的 OOD 检测任务上, 已有方法[10,36]训练辅助的对抗 OOD 样本来提升分布内边界对对抗扰动的鲁棒性. 然而, 由于辅助的 OOD 训练集与原 ID 训练集的分布差异, 仅训练干净 OOD 样本邻域内的对抗 OOD 样本并不能足够有效地使分布内边界对对抗扰动鲁棒, 即无法有效地阻止攻击者变异未见过的 OOD 样本在某分布内的类别上获取高的 Softmax 预测信心而再次绕过检测. 为了验证的这一关键见解, 我们采用公式 (4) 中的训练目标重新训练 DNN 并使用更强的由 Auto-PGD 优化的攻击来验证: 训练辅助的对抗 OOD 样本

是否能有效地使 OOD 检测对对抗扰动鲁棒? 我们使用 Auto-PGD 系列的强攻击来攻击训练过程中使用过的对抗 OOD 样本所对应的干净 OOD 样本以生成验证的对抗 OOD 样本. 这种验证的对抗 OOD 样本生成方法排除掉了验证的干净 OOD 样本与训练所使用的干净 OOD 样本之间的潜在分布差异的干扰, 降低了检测对抗 OOD 样本的难度. 如此一来, 如果所训练的检测器不能有效地检测这些验证的对抗 OOD 样本, 则说明仅训练对抗 OOD 样本对提升 OOD 检测的鲁棒性是不足的.

我们选择在 CIFAR10[43]上训练的 WRN-40-4[44]模型, 报告其平均 MSP (mean of MSP, MMSP) 分数以及 AUC 和 TPR-95 度量指标下的检测性能. 表 1 中, MMSP^{in} 表示在干净 ID 测试集上的 MMSP 分数, MMSP^{o} 表示在 OOD 样本上的 MMSP 分数, Clean^{o} 表示在辅助的干净 OOD 样本上的训练性能, PGD^{o} 表示在 (PGD 生成的) 对抗 OOD 样本上的训练性能, APGD^{o} 和 ACW^{o} 我们基于 Auto-PGD 开发的更强的攻击. 关于训练设置、度量指标以及 APGD^{o} 和 ACW^{o} 攻击的详细介绍在第 4.1 节给出. 如表 1 所示, ACET 和 ATOM 在 Clean^{o} 和 PGD^{o} 上的检测性能都接近完美, 证明当前训练在辅助的干净 OOD 样本和对抗 OOD 样本上已经良好地收敛. 然而, 当使用更强的 APGD^{o} 和 ACW^{o} 攻击相同的辅助 OOD 样本后, ACET 和 ATOM 的性能都出现了大幅度的下降, 甚至接近被完全攻破. 攻击后的更大的 MMSP^{o} 分数几乎无法用以区分干净 ID 样本和这些验证 OOD 样本. 该实验表明, 仅训练辅助的对抗 OOD 样本并不能足够有效地使分布内边界对对抗扰动鲁棒, 即无法有效地阻止攻击者变异 OOD 样本在 (原分布内) 某类别上获取高的 Softmax 预测信心来再次绕过检测. 在测试阶段, 对抗 OOD 样本一般是从与辅助的 OOD 样本存在潜在的分布差异的测试 OOD 样本上生成的, 检测这些未见过的恶意 OOD 样本将更具挑战性.

表 1 训练辅助的对抗 OOD 样本对检测验证的对抗 OOD 样本的性能 (↓表示越小越好, ↑表示越大越好)

方法	MMSP^{in}	Clean^{o}			PGD^{o}			APGD^{o}			ACW^{o}		
		MMSP^{o} (↓)	AUC (%) (↑)	TPR-95 (%) (↑)	MMSP^{o} (↓)	AUC (%) (↑)	TPR-95 (%) (↑)	MMSP^{o} (↓)	AUC (%) (↑)	TPR-95 (%) (↑)	MMSP^{o} (↓)	AUC (%) (↑)	TPR-95 (%) (↑)
ACET	0.9570	0.1063	99.56	99.88	0.1046	99.28	99.98	0.9090	14.77	10.70	0.8100	29.17	22.47
ATOM	0.9599	0.0004	99.51	100.0	0.0002	99.57	100.0	0.9999	2.06	0	0.9999	2.06	0

2.2 训练“近 OOD”样本——对抗 ID 样本

从干净 ID 样本的邻域内创建的对抗 ID 样本与干净 ID 样本享有近乎一样的语义信息, 是一种离分布内区域更“近”的 OOD 样本. 本节使用辅助的对抗 ID 样本而不使用任何辅助的对抗 OOD 样本来训练 DNN, 以调查其对提升 OOD 检测鲁棒性的作用.

$$\underset{\theta}{\operatorname{argmin}}\frac{1}{2N}\sum_{i=1}^{N}\left[\ell(f_\theta(x_i^{\text{in}}),y_i^{\text{in}})+\ell(f_\theta(x_i^{\text{in}}+\delta_i^{\text{in}*}),y_i^{\text{o}})\right]+\frac{1}{M}\sum_{j=1}^{M}\ell(f_\theta(x_j^{\text{o}}),y_j^{\text{o}}) \tag{5}$$

其中, N 和 M 分别表示干净 ID 样本和干净 OOD 样本的数量, $x_i^{\text{in}}+\delta_i^{\text{in}*}$ 表示从第 i 个 ID 样本创建的对抗 ID 样本, x_j^{o} 表示第 j 干净 OOD 样本, y^{o} 是带多个额外的拒绝类[11]的伪标签. $\delta_i^{\text{in}*}$ 的求解与公式 (3) 类似. 公式 (5) 为 $x_i^{\text{in}}+\delta_i^{\text{in}*}$ 标注了与 x_i^{in} 不同的伪标签 y_i^{o}, 这使得 DNN 能够更好地学习干净 ID 样本与其对抗 ID 样本的差异, 对扰动 $\delta_i^{\text{in}*}$ 建模. 需要注意的是公式 (5) 中对抗 ID 样本 $x_i^{\text{in}}+\delta_i^{\text{in}*}$ 也是在 DNN 每次的参数迭代步骤中依据当前阶段的 DNN 模型而实时生成的, 能更好地“覆盖”分布内每一类别的决策边界.

与表 1 中的结果相比, 表 2 中使用公式 (5) 所训练的模型在 APGD^{o} 和 ACW^{o} 强攻击下具备显著的性能优势, 即便其从未使用任何辅助的对抗 OOD 样本来训练 DNN. 该实验说明了训练对抗 ID 样本对提升分布内决策边界鲁棒性的作用同样是至关重要的, 在构建鲁棒的 OOD 检测器中不应被忽略.

表 2 训练对抗 ID 样本对检测 OOD 样本的影响

MMSP^{in}	Clean^{o}			PGD^{o}			APGD^{o}			ACW^{o}		
	MMSP^{o} (↓)	AUC (%) (↑)	TPR-95 (%) (↑)	MMSP^{o} (↓)	AUC (%) (↑)	TPR-95 (%) (↑)	MMSP^{o} (↓)	AUC (%) (↑)	TPR-95 (%) (↑)	MMSP^{o} (↓)	AUC (%) (↑)	TPR-95 (%) (↑)
0.9001	0.0272	99.20	97.53	0.0022	99.98	99.99	0.5299	**71.82**	**40.89**	0.5343	**71.55**	**40.46**

3 所提训练方法——谛听

在介绍谛听的训练目标之前, 本节首先介绍谛听中为分布外样本标注伪标签的方法以及所使用的用于区分 ID 样本和 OOD 样本的打分函数.

3.1 伪标签标注及打分函数

在检测干净 OOD 样本任务上, SSL[11]证明使用额外的多个拒绝类来表示 OOD 样本比使用均匀分布更具优势. 本文旨在同时检测干净 OOD 样本和对抗 OOD 样本, 这表明谛听所要应对的分布外样本更具多样性的特点. 鉴于此, 本文同样考虑为 DNN 分类器的最后一层添加多个拒绝类来表示分布外样本. 形式化地, 本文所考虑的带多个拒绝类的伪标签标注方法如下:

$$y^o=\begin{cases}\operatorname{argmax} f_\theta(x^{\text{clean}}), & \text{如果} \operatorname{argmax} f_\theta(x^{\text{clean}})>K\\ random[K,K+V], & \text{否则}\end{cases} \tag{6}$$

其中, x^{clean} 表示干净 ID 样本或干净 OOD 样本, K 表示原 ID 训练集中真实类别的数量, V 表示额外拒绝类的数量, $random[K,K+V]$ 表示取 [K, K+V] 内的随机整数 (此处假设类别索引从 1 开始编号, 由于谛听中辅助样本的伪标签是训练过程中自动标注的, 所以严格来讲谛听是一种自监督的对抗训练方法 (框架)). 对于对抗 ID 样本和对抗 OOD 样本, 我们使用它们的原干净 ID 样本和干净 OOD 样本来构建它们的伪标签. 使用干净 OOD 样本来为对抗 OOD 样本构造伪标签相当于鼓励 DNN 在 OOD 样本上习得对对抗扰动的不变性; 而为对抗 ID 样本标注不同于其原干净 ID 样本的伪标签则有利于 DNN 学习干净 ID 样本与其对抗 ID 样本的差异. 在第 4.4.3 节, 我们将实验性地验证使用多拒绝类来表示分布样本有利于增加攻击的难度.

关于检测/打分函数, 本文遵循文献 [6,15], 使用基于最大 Softmax 概率 (MSP) 的打分函数. 假设训练集包含样本的真实类别为 K, 则本文的 MSP 打分函数为:

$$D(x):\begin{cases}ID, & \text{如果} \max\left(f_\theta(x)_{[1:K]}\right)>\tau\\ \text{OOD}, & \text{否则}\end{cases} \tag{7}$$

其中, $f_\theta(x)_{[1:K]}$ 表示 $f_\theta(x)$ 前 K 维的真实类别内的最大 Softmax 预测概率, τ 是测试阶段指定的分数阈值. 对于 ID 样本的预测结果, 只需取前 K 类内最大 Softmax 预测概率的类别即可, 即 $\operatorname{argmax} f_\theta(x^{\text{in}})_{[1:K]}$. 此外, 其他更先进的打分函数也可以应用于本文所训练的模型上以获得更好的性能.

3.2 训练目标

在第 2.2 节, 我们的实验结果验证了训练辅助的对抗 ID 样本作为 OOD 样本对提升分布内边界鲁棒性的有效性. 在谛听中, 我们考虑同时训练辅助的对抗 ID 样本、干净 OOD 样本以及对抗 OOD 样本:

$$\underset{\theta}{\operatorname{argmin}}\frac{1}{2N}\sum_{i=1}^{N}\left[\ell(f_\theta(x_i^{\text{in}}),y_i^{\text{in}})+\ell(f_\theta(x_i^{\text{in}}+\delta_i^{\text{in}*}),y_i^{\text{o}})\right]+\beta\cdot\frac{1}{M}\left[\sum_{j=1}^{M}\ell(f_\theta(x_j^{\text{o}}),y_j^{\text{o}})+\ell(f_\theta(x_j^{\text{o}}+\delta_j^{\text{o}*}),y_j^{\text{o}})\right] \tag{8}$$

其中, N 表示原 ID 训练集样本的总数量, M 表示辅助 OOD 样本的数量, β 用于控制原 ID 训练数据 (即干净 ID 样本及其对抗 ID 样本) 和辅助 OOD 数据 (即干净 OOD 样本和对抗 OOD 样本) 的均衡, 伪标签 y_i^{o} 和 y_j^{o} 都是根据公式 (6) 构建的, 其他符号的含义与公式 (4) 和公式 (5) 中的一致.

公式 (8) 中的第 1 项为对抗 ID 样本 $x_i^{\text{in}}+\delta_i^{\text{in}*}$ 标注了不同于 x_i^{in} 的伪标签 y_i^{o}, 用以使 DNN 学习干净 ID 样本与其对抗 ID 样本之间的差异; 第 2 项中, 使用干净 OOD 样本来为对抗 OOD 样本的构建伪标签并把它们同时输入 DNN, 使得 DNN 在 OOD 样本上更好地习得对对抗扰动的不变性; 同时, 干净 OOD 样本和对抗 OOD 样本的伪标签使得 DNN 在它们上习得相对于 ID 数据的不同. 结合公式 (8) 的第 1 项和第 2 项可以看到, 其最终使 DNN 对分布外的样本 (即对抗 ID 样本、干净 OOD 样本和对抗 OOD 样本) 习得不同于干净 ID 样本的统一的“认知”. 与公式 (3) 中的常规对抗训练相比, 公式 (8) 中 $x_i^{\text{in}}+\delta_i^{\text{in}*}$ 的伪标签 y_i^{o} 并不会强迫 DNN 完全忽略那些与真实标签弱相关的特征, 即与 $\delta^{\text{in}*}$ “类似”的、难以察觉的特征来做决策. 因此, 公式 (8) 并不会导致 DNN 在干净 ID 样本上的分类

准确率显著下降, 且能同时使得分布内边界邻域内的扰动可被检测.

3.3 对抗扰动搜索策略

搜索对抗扰动是求解 AT 的内部 max 的一个关键步骤. 例如, 在常规的 AT 中, 训练由 PGD 攻击搜索的对抗样本能比由有目标的 PGD 攻击和 CW 攻击生成的对抗样本获得更好的鲁棒性[42]. 因此, 如何搜索公式 (8) 中的对抗扰动 $\delta_i^{\text{in}*}$ 和 $\delta_j^{\text{o}*}$ 同样至关重要. 对于公式 (8) 中的从 ID 扰动生成的对抗 ID 扰动 $\delta_i^{\text{in}*}$, 本文遵循常规的 AT, 使用 PGD 攻击来最大化输入 $x_i^{\text{in}}+\delta_i^{\text{in}}$ 及其真实标签 y_i^{in} 上的分类损失以近似地搜索最优扰动: $\delta_i^{\text{in}*}=\text{argmax}_{\|\delta_i^{\text{in}}\|_p\leqslant\epsilon}\ell(f_\theta(x_i^{\text{in}}+\delta_i^{\text{in}}),y_i^{\text{in}})$. 对于公式 (8) 中的对抗 OOD 扰动 $\delta_j^{\text{o}*}$, 本文通过最小化 f_θ 在 V 个拒绝类上的最大 Softmax 预测信心来创建扰动, 所对应的对抗损失如下:

$$\ell\left(f_\theta\left(x_j^{\text{o}}+\delta_j^{\text{o}}\right),\text{argmax}\left(f_\theta\left(x_j^{\text{o}}+\delta_j^{\text{o}*}\right)_{[K+1:K+V]}\right)\right)=-\log\left(\max\left(f_\theta\left(x_j^{\text{o}}+\delta_j^{\text{o}}\right)_{[K+1:K+V]}\right)\right) \tag{9}$$

其中, $f_\theta\left(x_j^{\text{o}}+\delta_j^{\text{o}*}\right)_{[K+1:K+V]}$ 表示 f_θ 在最后 V 个拒绝类上的 Softmax 预测输出. 通过使用 PGD 求解 $\delta_j^{\text{o}*}=\text{argmax}_{\|\delta_j^{\text{o}}\|_p\leqslant\epsilon}-\log(\max(f_\theta\left(x_j^{\text{o}}+\delta_j^{\text{o}}\right)_{[K:K+V]}))$, 可以近似地得到能使 f_θ 在额外拒绝类上输出最小预测信心的对抗 OOD 样本 $x_j^{\text{o}}+\delta_j^{\text{o}*}$. 另一种直观的搜索对抗 OOD 扰动的策略是直接最大化 OOD 样本的 MSP 分数. 这种策略相当于把 OOD 样本在前 K 类内具有最大预测信心的类, 作为攻击目标, 是一种有目标的攻击, 所获得的扰动不具备“普适性”, 并不能很好地使 OOD 检测器鲁棒. 本文将在第 4.3.3 节的消融研究中对比这两种策略对 OOD 检测器鲁棒性的影响.

4 实验分析

遵循鲁棒性研究领域主流的设置, 本文同样考虑 L_∞ Norm 约束下的攻击. 接下来, 我们在第 4.1 节介绍实验设置, 在第 4.2 节中提供主要实验结果与分析, 最后在第 4.3 节中进行消融实验.

4.1 实验设置

4.1.1 数据集

关于分布内数据集、辅助的分布外训练集和分布外测试集, 本文遵循大多数工作[6,11,15,16]中的主流设置, 所选择的开源数据集如下.

(1) 分布内数据集. 本文选择 SVHN[45]、CIFAR10 和 CIFAR100[43]这 3 种数据集作为分布内数据集. 其中, SVHN 是一个包含 0–9 数字的门牌号数据集, 其训练集和测试集分别包含 73 257 张和 26 032 张 32×32 的彩色图片; CIFAR10 由真实世界中 10 种不同的物种组成, 训练集和测试集分别有 50 000 张和 10 000 张 32×32 彩色图像; CIAFR100 与 CIFAR10 类似, 所不同的是 CIFAR100 包含 100 类 60 000 张图片.

(2) 辅助的分布外训练集. 本文使用包含 8 000 万张 32×32 的彩色图片的 80 Million Tiny Images[46]作为辅助的分布外训练集. 该数据集在防御和检测等研究领域中被大多数的半监督训练方法广泛采用. 在训练时, 本文遵循[6,11,15,16]的方法, 同样把 80 Million Tiny Images 中与分布内数据集“雷同”的数据排除掉 (先前的工作已经提供了需要排除的 id 列表, 我们只需要依据该列表相应地排除辅助样本即可).

(3) 分布外测试集. 默认情况下, 本文选择 5 种分布外测试集: Places365[47]、Textures[48]、iSUN[49]、LSUN (crop) 和 LSUN (resize)[50]. 如果分布内训练集是 SVHN, 本文将 CIFAR10 和 CIFAR100 的测试集视作分布外测试集; 以此类推, 如果分布内训练集是 CIFAR10 或 CIFAR100, 本文同样将 SVHN 的测试集视作分布外测试集. 本文混合这 6 或 7 种分布外测试集来构建一个混合的分布外测试集, 用以模拟真实世界中的分布外样本数据. 在接下来的实验中, 如无特别说明, 本文默认报告检测器在该混合的分布外测试集上的检测性能.

4.1.2 训练设置

遵循文献 [16], 本文选择 WRN-40-4[44]模型并使用 SGD 优化器来执行训练. SGD 优化器的设置如下: 动量 0.9, 权重衰减 0.000 5, 初始学习率 0.1. 在所有训练集上, 我们将超参 β 设为 1, 并将 ID 数据和辅助的 OOD 数据的 batch size 都设为 128. 额外拒绝类的数量 V 在 SVHN、CIFAR10 和 CIFAR100 上分别设置为 4、10 和 35. 在 SVHN 上, 我们训练 50 个 epochs, 并在第 25 和第 40 个 epoch 时将学习率分别除以 10; 在 CIFAR10 和 CIFAR100 上,

我们训练 200 个 epochs, 并在第 150 和第 180 个 epoch 时将学习率衰减为此前的 1/10. 对于训练和测试期间允许扰动的最大扰动半径 ϵ, 我们同样遵循鲁棒性研究领域的主流设置[42], 将其设置为 8/255. 在训练阶段, 我们使用攻击步长为 1/255 的 PGD-10 和 PGD-20 攻击来分别搜索对抗性 ID 样本和对抗性 OOD 样本. 在测试阶段, 我们使用 PGD-20 和 Auto-PGD 来生成对抗 OOD 样本 (Auto-PGD 的重启次数以及迭代轮数等超参数遵循其默认设置). 此外, 本文在训练对抗 ID 数据上引入了热启动策略, 在 SVHN 的第 10 个 epoch 以及 CIFAR10 和 CIFAR100 的第 100 个 epoch 后才加入训练对抗 ID 样本. 其他未列出的设置则遵循文献 [15,16].

4.1.3 用于生成测试对抗 OOD 样本的攻击

在谛听中, DNN 分类器同时兼做 OOD 检测器. 因此, 可以通过直接修改那些原用于攻击分类器的攻击来攻击 OOD 检测器, 即最大化前 K 个真实类内的最大 Softmax 预测概率 (MSP) 和最大 logit 即可. 具体来讲, 本文首先考虑使用交叉熵 (CE) 损失和 CW 损失以及 PGD 搜索算法和 Auto-PGD 搜索算法来组合不同强度的无目标攻击. 使用 CE 损失最大化前 K 类内的最大 Softmax 预测概率 (MSP) 的攻击如下:

$$-\ell\left(f_\theta(x^o+\delta^o), \operatorname{argmax}\left(f_\theta(x^o+\delta^{o*})_{[0:K]}\right)\right)=\log\left(\max\left(f_\theta(x^o+\delta^o)_{[1:K]}\right)\right) \tag{10}$$

其中, $\operatorname{argmax}(f_\theta(x^o+\delta^{o*})_{[1:K]}$ 表示前 K 个真实类别内具有最大 MSP 的类别. 本文把由 PGD 和 Auto-PGD 求解的关于公式 (10) 的攻击分别记为 PGD^o 和 APGD^o. 对于使用 CW 的变种损失，最大化前 K 类内的最大 logit 的攻击的损失为:

$$\max_k\left(z_\theta(x^o+\delta^o)_{[0:K]}\right)-\max_{k\neq j}\left(z_\theta(x^o+\delta^o)_k\right) \tag{11}$$

其中, 第 1 项表示前 K 类内的最大 logit, 第 2 项表示整个 logits 层中除了第 1 项之外最大的 logit. 本文同样分别使用 PGD 搜索算法和 Auto-PGD 搜索算法来求解公式 (11), 并把它们分别记为 CW^o 和 ACW^o.

其次, 本文还考虑多目标的攻击, 以进一步攻破 OOD 检测器. 有目标的 APGD^o 攻击为: $\log\left(f_\theta(x^o+\delta^o)_t\right)$, 其中 $1\leqslant t<K$; 对于有目标的 ACW^o 攻击, 直接将公式 (10) 中的第 1 项替换为 $z_\theta(x^o+\delta^o)_t$ 即可. 为方便区分, 本文将有目标的 APGD^o 和有目标的 ACW^o 分别记为 APGD^t 和 ACW^t. 在实际攻击时, 本文将 t 分别设定为前 9 个 Softmax 预测信心最大的类别, 执行 9 次具有不同目标 t 的攻击, 并返回具有最大的 MSP 分数的样本.

4.1.4 评价指标

在 OOD 样本检测任务中, OOD 样本一般视为正样本, ID 样本视为负样本. 本文考虑如下度量指标.

(1) ROC 曲线下面积 (area under the ROC, AUC). 该指标无关于打分函数的不同阈值 τ, 反映了分不外样本检测的整体性能. 越大的 AUC 值表示 OOD 检测器的整体性能越好.

(2) TPR at TNR-N (TPR-N). 该指标反映在错误地将 (100 – N)% 的 ID 样本识别为 OOD 样本的情况下 (即 N% 的真阴性率) 对真正的 OOD 样本检测的正确率. TPR-N 是一种只依赖于干净 ID 测试集的具体性能指标. 当使用 TPR-N 指标时, 打分函数的阈值 τ 是由 ID 样本决定的, 与待检测的 OOD 数据集无关. 越大的 TPR-N 意味着 OOD 检测器的性能越好.

此外, 本文报告攻击前后的平均 MSP (MMSP) 分数, 以方便直观地观察攻击前后 OOD 样本的 MSP 分数的变化.

4.2 主要实验结果与分析

本文主要考虑前沿的 SSL[11]、ACET[10]、ATOM[36]和 AT$^{\text{in}}_{\text{out}}$ [12,13]作为对比对象. 其中, SSL 的多个拒绝类的伪标签与本文所采用的伪标签类似, 但其未考虑任何对抗样本; 后 3 种方法都引入了对抗训练来构建鲁棒的 OOD 检测器, 且在检测 OOD 样本任务上取得了前沿的鲁棒性能.

4.2.1 混合的分布外测试集上的性能

与鲁棒性研究领域类似, 本文同样关注 OOD 检测器在不同攻击下的最差性能. 同时, DNN 模型在原主任务上的性能 (即在干净 ID 样本上的分类准确率) 以及在干净 OOD 样本的检测性能同样重要, 不应被显著地牺牲. 实验结果如表 3 所示 (比较参数均为越大越好), 其中, 显著优于其它方法的结果被加粗, D^{in} 代表分布内训练集, Acc 代表 DNN 在干净 ID 样本上的分类准确率, Ours 代表本文所提的谛听. 此外, 由于页面空间限制, 本节把每种检测方法攻击前后的 MMSP 分数汇总在表 4 中. 总体来讲, 谛听在不同攻击下的最差检测性能显著优于其他已有方法, 并且在分类干

净 ID 样本任务上和检测干净 OOD 样本任务上保持先进的性能. 与 SSL、ACET 和 ATOM 相比, 谛听在不显著损害原分类性能及检测干净 OOD 样本性能前提下, 在检测由强攻击生成的对抗 OOD 样本上取得明显甚至是压倒性的性能优势; 与 AT^{in}_{out} 相比, 谛听在原分类任务、检测干净 OOD 样本及检测对抗 OOD 样本上全面取得显著的优势.

表 3 检测 OOD 样本的性能 (TPR-*N* 在 SVHN 和 CIFAR10 上指 TPR-95; 在 CIFAR100 上指 TPR-80) (%)

D^{in}	Method	Acc	Clean^o		PGD^o		CW^o		APGD^o		ACW^o		APGD^t		ACW^t	
			AUC	TPR-*N*	AUC	TPR-*N*	AUC	TPR-*N*	AUC	TPR-*N*	AUC	TPR-*N*	AUC	TPR-*N*	AUC	TPR-*N*
SVHN	SSL	96.42	99.97	99.89	38.87	18.67	39.36	18.66	22.85	9.54	23.06	9.60	20.36	7.97	20.44	8.02
	ACET	96.39	99.95	99.78	99.92	99.68	99.95	99.78	91.19	81.68	91.55	82.48	88.13	75.00	88.42	75.57
	ATOM	96.49	99.99	99.99	99.98	99.98	99.99	99.98	83.12	67.46	83.44	67.43	76.58	59.21	76.93	59.23
	AT^{in}_{out}	94.13	99.62	93.38	94.88	77.90	95.52	79.84	94.46	76.59	94.91	77.68	94.01	75.22	94.48	76.26
	Ours	96.75	99.98	99.92	99.97	99.92	99.97	99.92	**98.12**	**92.88**	**98.14**	**93.06**	**98.53**	**93.97**	**98.53**	**94.22**
CIFAR10	SSL	95.21	99.04	97.04	52.29	8.84	0.98	0.07	0.04	0.00	0.04	0.00	0.03	0.00	0.03	0.00
	ACET	95.87	98.72	96.06	98.13	95.07	98.19	95.17	9.31	6.42	21.68	15.93	1.86	0.68	2.68	1.11
	ATOM	95.97	99.34	98.49	97.83	91.46	97.84	91.30	0.29	0.05	0.32	0.05	0.26	0.05	0.26	0.05
	AT^{in}_{out}	86.73	94.00	66.80	80.73	24.14	82.28	27.71	80.18	22.98	81.16	24.88	78.62	19.56	79.65	20.77
	Ours	94.57	98.89	96.62	97.81	96.62	97.81	96.62	**96.89**	**95.13**	**97.17**	**95.46**	**94.18**	**78.84**	**95.07**	**84.15**
CIFAR100	SSL	77.55	91.04	87.84	38.19	12.05	0.48	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	ACET	77.90	89.24	83.50	89.17	83.48	89.18	83.48	7.78	0.80	15.30	3.29	3.44	0.11	8.57	0.56
	ATOM	78.98	93.36	93.05	91.27	85.08	91.27	85.09	0.57	0.07	0.06	0.08	0.22	0.05	0.23	0.05
	AT^{in}_{out}	60.95	79.92	59.56	51.30	9.56	56.30	14.63	50.45	8.82	54.18	11.68	48.44	7.11	52.03	9.10
	Ours	75.97	89.79	85.37	88.82	85.34	88.84	85.33	**79.17**	**71.76**	**80.71**	**73.80**	**80.72**	**72.95**	**81.77**	**75.35**

表 4 平均最大 Softmax 分数 (MMSP)

D^{in}	Method	ID	Clean^o	PGD^o	CW^o	APGD^o	ACW^o	APGD^t	ACW^t
SVHN	SSL	0.9835	0.0065	0.8842	0.8841	0.9438	0.9437	0.9547	0.9547
	ACET	0.9875	0.1153	0.1168	0.1168	0.4217	0.4115	0.5117	0.5066
	ATOM	0.9850	0.0018	0.0019	0.0019	0.4634	0.4641	0.5487	0.5489
	AT^{in}_{out}	0.8548	0.1243	0.2590	0.2434	0.2697	0.2594	0.2817	0.2723
	Ours	0.9878	0.0063	0.0103	0.0102	0.2087	0.2055	0.2076	0.2036
CIFAR10	SSL	0.9208	0.0265	0.8742	0.9988	0.9999	0.9999	0.9999	0.9999
	ACET	0.9570	0.1477	0.1649	0.1627	0.9469	0.8686	0.9945	0.9910
	ATOM	0.9593	0.0262	0.1099	0.1105	0.9995	0.9995	0.9996	0.9995
	AT^{in}_{out}	0.7446	0.2285	0.4157	0.3942	0.4235	0.4098	0.4455	0.4310
	Ours	0.9294	0.0308	0.0327	0.0327	0.0480	0.0444	0.1877	0.1525
CIFAR100	SSL	0.7616	0.1260	0.8587	0.9995	1.0000	1.0000	1.0000	1.0000
	ACET	0.7976	0.2129	0.2135	0.2134	0.9885	0.9618	0.9982	0.9921
	ATOM	0.8534	0.2042	0.2115	0.2115	0.9993	0.9993	0.9995	0.9995
	AT^{in}_{out}	0.4665	0.1497	0.4167	0.3588	0.4267	0.3806	0.4499	0.4028
	Ours	0.7487	0.1397	0.1403	0.1402	0.2745	0.2541	0.2262	0.2518

不使用任何对抗 OOD 样本训练的 SSL 在弱的 PGD 系列攻击 (即 PGD^o 和 CW^o) 下即被大幅度地击败. ACET 和 ATOM 在检测 PGD 系列的攻击生成的对抗 OOD 样本上都取得先进的性能, 且几乎没有损害原分类任务和检测干净 OOD 样本的性能. 然而, 在检测由更强的 Auto-PGD 系列的攻击 (即 APGD^o、ACW^o、APGD^t 和 ACW^t) 生成的对抗 OOD 样本上, 它们的性能都出现了大幅度地下降, 在 CIFAR 系列数据集上几乎被完全攻破. 这证明了仅训练辅助的对抗 OOD 样本无法有效地使分布内决策边界对对抗扰动足够鲁棒, 而只是呈现了针对 PGD 系列攻击的"过拟合", 或者说仅呈现了梯度混淆[38]的虚假安全. 在 ID 数据上引入常规对抗训练 (AT) 的 AT^{in}_{out} 虽然相较于 ACET 和 ATOM 有效地提升了 OOD 检测的鲁棒性, 但是却显著地损害了对干净 ID 样本的分类准确率以及检测干净 OOD 样本的性能; 其在 TPR-*N* 指标下的结果相较于谛听也较不理想, 这可能是由于其在干净 ID

样本和干净 OOD 样本上相对较差的性能引起的. 此外, AT^{in}_{out} 在检测 PGD 系列的攻击产生的对抗 OOD 样本的性能也明显差于 ACET、ATOM 和谛听. 相较于以上所有方法, 谛听克服了它们存在的不足, 不仅使用辅助的干净 OOD 样本和对抗的 OOD 样本, 也使用对抗 ID 样本作为辅助的 OOD 样本来训练 DNN, 这使得谛听在几乎不损害原分类任务性能和检测干净 OOD 样本性能的前提下, 在保证 OOD 检测器鲁棒性任务上取得显著性的甚至是压倒性的优势. 在表 4 中, 我们同样可以看到谛听的 $MMSP^{o}$ 在各种攻击后与其 $MMSP^{in}$ 的差异是最显著的, 更易于区分 OOD 样本和 ID 样本.

4.2.2 不同的单个分布外测试集上的性能

第 4.2.1 节报告了谛听在由 6–7 种分布外测试数据集上的鲁棒性能, 本节进一步报告其在不同的单个分布外测试集上的性能. 所选择的分布内数据集和模型分别是 CIFAR10 及 WRN-40-4. 实验结果如表 5 所示 (比较参数均为越大越好), D^{o} 表示各单独的分布外测试集的名称, 其他各指标的含义与表 3 相同. 从表 5 的实验结果可以看出, 谛听在每一种单独的分布外测试集上相较于已有方法均显示出了显著的鲁棒性能优势.

表 5 在不同的单个分布外测试数据集上的检测性能 (%)

D^{o}	Method	$Clean^{o}$		PGD^{o}		CW^{o}		$APGD^{o}$		ACW^{o}		$APGD^{t}$		ACW^{t}	
		AUC	TPR-95	AUC	TPR-95	AUC	TPR-95	AUC	TPR-95	AUC	TPR-95	AUC	TPR-95	AUC	TPR-95
Places365	SSL	97.45	91.16	61.15	07.29	0.10	0	0	0	0	0	0	0	0	0
	ACET	96.90	90.15	96.58	90.15	96.60	90.15	5.87	3.80	15.04	10.43	0.65	0.17	0.62	0.19
	ATOM	97.41	91.79	97.04	91.79	97.04	91.78	0	0	0	0	0	0	0	0
	AT^{in}_{out}	90.16	52.42	72.12	10.05	73.91	12.87	71.50	9.20	72.54	10.39	69.45	6.39	70.57	7.00
	Ours	97.55	92.37	96.55	92.37	96.55	92.37	**96.16**	**91.70**	**96.27**	**91.85**	**94.42**	**81.05**	**95.33**	**86.52**
SVHN	SSL	99.09	97.25	80.20	61.70	0	0	0	0	0	0	0	0	0	0
	ACET	99.14	97.74	98.75	97.74	98.84	97.74	6.35	3.45	23.69	16.28	1.38	0.09	1.43	0.09
	ATOM	99.87	99.53	99.37	99.53	99.37	99.53	0	0	0	0	0	0	0	0
	AT^{in}_{out}	95.21	67.79	81.61	17.46	83.91	23.00	80.89	16.36	82.27	18.92	78.84	12.68	80.29	14.30
	Ours	99.21	97.44	98.07	97.44	98.07	97.44	**98.03**	**97.40**	**98.04**	**97.40**	**97.80**	**95.66**	**98.00**	**97.21**
LSUN (crop)	SSL	99.50	99.50	59.48	9.66	3.92	0.26	0	0	0	0	0	0	0	0
	ACET	99.52	99.07	99.14	99.07	99.20	99.07	11.80	7.66	31.46	23.00	3.07	0.78	2.60	0.72
	ATOM	99.57	99.39	99.38	99.39	99.38	99.39	0.22	0	0.23	0	0.11	0	0.05	0
	AT^{in}_{out}	98.10	88.91	90.32	52.87	91.32	56.35	89.79	51.08	90.37	53.17	88.56	47.05	89.20	48.57
	Ours	99.56	99.29	98.38	99.29	98.38	99.29	**98.33**	**99.25**	**98.33**	**99.26**	**98.19**	**98.23**	**98.25**	**98.80**
LSUN (resize)	SSL	99.63	99.17	28.32	0.32	0.08	0	0	0	0	0	0	0	0	0
	ACET	99.10	97.46	98.51	95.76	98.50	95.91	9.70	7.58	16.21	12.97	0.42	0.19	0.33	0.13
	ATOM	99.83	99.64	96.99	84.99	96.98	84.25	0.08	0	0.11	0	0.05	0	0.06	0
	AT^{in}_{out}	93.87	66.78	79.71	19.74	80.94	23.04	79.27	18.67	80.09	20.43	77.97	14.61	78.83	15.72
	Ours	99.16	97.44	98.10	97.44	98.10	97.44	**96.14**	**93.90**	**96.73**	**94.61**	**89.92**	**56.36**	**91.45**	**65.40**
iSUN	SSL	99.62	99.01	28.65	0.88	0.18	0	0.02	0	0.02	0	0	0	0	0
	ACET	98.93	96.32	97.48	92.14	97.64	92.64	8.87	6.72	15.26	11.84	0.24	1.34	0.27	0.06
	ATOM	99.78	99.49	95.50	76.68	95.58	76.53	0.25	0	0.34	0	0.15	0	0.20	0
	AT^{in}_{out}	92.00	58.52	76.82	13.87	78.18	16.76	76.36	12.97	77.32	14.70	75.09	9.75	76.05	10.82
	Ours	99.14	97.43	98.09	97.43	98.09	97.43	**95.31**	**93.49**	**96.21**	**94.41**	**88.64**	**55.21**	**90.72**	**66.10**
Textures	SSL	99.05	95.81	54.22	10.19	2.06	0.30	0.36	0.03	0.35	0.03	0.28	0.03	0.30	0.03
	ACET	98.74	95.39	98.31	95.33	98.33	95.28	16.24	11.56	32.38	24.27	7.79	4.18	7.74	4.23
	ATOM	99.53	98.03	98.94	97.37	98.95	97.39	1.89	0.49	1.97	0.53	1.97	0.44	2.01	0.49
	AT^{in}_{out}	94.82	64.48	85.41	34.11	87.11	37.19	84.91	32.89	86.13	35.03	83.65	30.70	84.81	32.12
	Ours	98.68	95.26	97.66	95.27	97.66	95.27	**97.39**	**94.66**	**97.47**	**94.87**	**96.50**	**87.93**	**97.07**	**92.66**

4.3 消融实验

本节研究不同额外拒绝类数量、单独取消训练对抗 ID 样本或者对抗 OOD 样本以及使用其他对抗 OOD 样本搜索策略对 OOD 检测器鲁棒性的影响. 所选择数据集和模型分别是 CIFAR10 和 WRN-40-4.

4.3.1 额外拒绝类的数量

多个额外的拒绝类表达了分布外空间的多样性, 本节固定其他所有的设置, 调查不同的 V 的设置对 OOD 检测器鲁棒性的影响. 实验结果如表 6 所示 (比较参数均越大越好), 当 V 为 10 时, OOD 检测器取得最好的鲁棒性能, 过小或过大的 V 都不能取得最优的鲁棒性能. SSL 声称适当多的拒绝类对提升检测干净 OOD 样本的性能有利, 本实验进一步证实适当多的拒绝类对提升 OOD 检测的鲁棒性同样有利.

表 6 变动拒绝类数量对检测对抗 OOD 样本的影响 (%)

V	Clean^o		PGD^o		CW^o		APGD^o		ACW^o		APGD^t		ACW^t	
	AUC	TPR-95	AUC	TPR-95	AUC	TPR-95	AUC	TPR-95	AUC	TPR-95	AUC	TPR-95	AUC	TPR-95
5	98.81	97.01	97.63	97.01	97.62	97.01	84.84	64.84	87.16	69.16	86.09	61.10	87.83	65.60
10	98.89	96.62	97.81	96.62	97.81	96.62	**96.89**	**95.13**	**97.17**	**95.46**	**94.18**	**78.84**	**95.07**	**84.15**
15	98.88	96.23	98.12	96.23	98.13	96.22	85.25	71.50	87.98	75.64	87.78	69.51	90.77	78.23

4.3.2 取消训练对抗 ID 样本或对抗 OOD 样本

本节首先取消训练对抗 OOD 样本, 保持其他设置不变, 以验证训练辅助的对抗 ID 样本作为 OOD 样本对提升分布内决策边界的鲁棒性的作用. 然后, 我们只取消训练对抗 ID 样本, 保持其他设置不变, 以验证仅训练辅助的对抗 OOD 样本对带有多拒绝类的检测器鲁棒性的影响.

实验结果如表 7 所示 (比较参数均越大越好), 其中, Org.指谛听, “−ADV OOD”表示在谛听的基础上取消训练对抗 OOD 样本, “−ADV ID”指在谛听的基础上取消训练 ID 样本. 当取消训练对抗 OOD 样本后, “−ADV OOD”在 Auto-PGD 攻击下的检测性能虽然出现较明显的下降, 但是依然未被完全攻破, 这证明了训练对抗 ID 样本作为辅助的 OOD 样本对提升分布内边界鲁棒性的有效性. 取消训练对抗 ID 样本后, 谛听的训练目标变得与 ACET 和 ATOM 类似, 所不同的是它的伪标签是带多个拒绝类的伪标签. 对比“−ADV ID”与表 3 中的 ACET 和 ATOM, “−ADV ID”在 Auto-PGD 系列攻击下的性能依然显著好于它们, 这证明了使用多个拒绝边界比使用单个拒绝边界 (例如 ATOM) 或者为分布外样本分配均匀分布 (例如 ACET) 对提升分布内边界的鲁棒性更有效.

表 7 取消训练对抗 ID 样本或对抗 OOD 样本对 OOD 检测鲁棒性的影响 (%)

Method	Clean^o		PGD^o		CW^o		APGD^o		ACW^o		APGD^t		ACW^t	
	AUC	TPR-95	AUC	TPR-95	AUC	TPR-95	AUC	TPR-95	AUC	TPR-95	AUC	TPR-95	AUC	TPR-95
Org.	98.89	96.62	97.81	96.62	97.81	96.62	**96.89**	**95.13**	**97.17**	**95.46**	**94.18**	**78.84**	**95.07**	**84.15**
−ADV OOD	98.92	96.15	98.91	96.09	98.90	96.05	62.54	29.82	63.97	43.38	47.96	14.6	59.2	25.0
−ADV ID	99.16	97.02	99.11	96.97	99.11	96.97	58.30	42.24	58.53	42.16	47.42	27.14	48.27	29.71

4.3.3 变更对抗 OOD 样本搜索策略

在谛听训练过程中, 我们通过最小化多个额外的拒绝类的最大 Softmax 预测概率来创建对抗 OOD 样本. 在本节中, 我们最大化 OOD 样本在前 K 类内的最大 Softmax 预测概率 (MSP 分数) 来搜索对抗扰动, 以研究其对 OOD 检测鲁棒性的影响. 具体而言, 我们通过使用 PGD 攻击优化 $\delta_j^{o*}=\mathrm{argmax}_{\|\delta_j^o\|_p\leqslant\epsilon}\log\left(\max\left(f_\theta\left(x_j^o+\delta_j^o\right)_{[1:K]}\right)\right)$ 来搜索对抗扰动, 其中 $f_\theta\left(x_j^o+\delta_j^o\right)_{[1:K]}$ 表示 $x_j^o+\delta_j^o$ 在前 K 类内的最大 Softmax 预测信心.

实验结果如表 8 所示 (比较参数均越大越好), 其中 Max-MSP 表示最大化 MSP 分数搜索对抗扰动. 在无目标的 Auto-PGD 系列攻击下, Max-MSP 与原谛听的检测性能差距不是很大; 但当使用多目标的 APGD^t 和 ACW^t 攻击后, Max-MSP 的检测性能出现了明显下降. 直接最大化 OOD 的 MSP 分数可以视为一种有目标的攻击, 其攻击目标是当前 OOD 样本在前 K 类内具有最大预测信心的类别. 训练此类扰动可以有效阻止无目标的 APGD^o 和 ACW^o 攻击, 因为它们同样是基于当前 OOD 样本的前 K 类内具有最大 Softmax 预测概率的类别来发起攻击的. 然而, 当使用 APGD^t 和 ACW^t 攻击时, 它们的攻击目标轮流设置为其他非最大 Softmax 预测概率的类别, Max-MSP 的性能会出现显著下降.

表 8 使用其他 OOD 扰动搜索策略对 OOD 检测器鲁棒性的影响 (%)

Method	Clean^o		PGD^o		CW^o		APGD^o		ACW^o		APGD^t		ACW^t	
	AUC	TPR-95	AUC	TPR-95	AUC	TPR-95	AUC	TPR-95	AUC	TPR-95	AUC	TPR-95	AUC	TPR-95
Org.	98.89	96.62	97.81	96.62	97.81	96.62	**96.89**	**95.13**	**97.17**	**95.46**	**94.18**	**78.84**	**95.07**	**84.15**
Max-MSP	98.84	96.45	98.02	96.44	98.05	96.44	94.35	88.52	91.87	85.58	83.92	67.72	89.38	77.91

5 总 结

检测干净 OOD 样本和带恶意扰动的对抗 OOD 样本对 DNN 模型在开放环境下的部署的至关重要. 由于辅助的 OOD 训练集与原 ID 训练集的分布差异, 仅训练辅助的对抗 OOD 样本不能足够有效地使分布内边界对对抗扰动足够鲁棒. 从干净 ID 样本的邻域内创建的对抗 ID 样本是一种离分布内区域更近的 OOD 样本. 本文首先实证了训练辅助的对抗 ID 样本作为分布外样本对提升分布内决策边界鲁棒性的有效性, 然后提出了一种半监督的对抗训练方法——谛听来提升 OOD 检测的鲁棒性. 谛听把对抗 ID 样本视为“近 OOD”样本, 同时使用辅助的对抗 ID 样本、干净 OOD 样本和对抗 OOD 样本来联合训练 DNN. 实验结果表明, 谛听在不显著损害原分类性能的前提下, 在检测由更强的攻击产生的对抗 OOD 样本上具备显著的性能优势, 并在检测干净 OOD 样本上保持先进的性能. 消融实验进一步表明谛听使用额外的多拒绝类来表示分布外样本同样有利于提升 OOD 检测的鲁棒性.

References:

[1] He KM, Zhang XY, Ren SQ, Sun J. Deep residual learning for image recognition. In: Proc. of the 2016 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR). Las Vegas: IEEE, 2016. 770–778. [doi: 10.1109/CVPR.2016.90]

[2] Bojarski M, Del Testa D, Dworakowski D, Firner B, Flepp B, Goyal P, Jackel LD, Monfort M, Muller U, Zhang JK, Zhang X, Zhao J, Zieba K. End to end learning for self-driving cars. arXiv:1604.07316, 2016.

[3] Chen JN, Lu YY, Yu QH, Luo XD, Adeli E, Wang Y, Lu L, Yuille AL, Zhou YY. TransUNet: Transformers make strong encoders for medical image segmentation. arXiv:2102.04306, 2021.

[4] Hendrycks D, Gimpel K. A baseline for detecting misclassified and out-of-distribution examples in neural networks. arXiv:1610.02136, 2018.

[5] Hendrycks D, Gimpel K. Early methods for detecting adversarial images. arXiv:1608.00530, 2017.

[6] Hendrycks D, Mazeika M, Dietterich T. Deep anomaly detection with outlier exposure. arXiv:1812.04606, 2019.

[7] Feinman R, Curtin RR, Shintre S, Gardner AB. Detecting adversarial samples from artifacts. arXiv:1703.00410, 2017.

[8] Lee K, Lee K, Lee H, Shin J. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In: Proc. of the 32nd Int'l Conf. on Neural Information Processing Systems. Montréal: Curran Associates Inc., 2018. 7167–7177.

[9] Lee K, Lee H, Lee K, Shin J. Training confidence-calibrated classifiers for detecting out-of-distribution samples. arXiv:1711.09325, 2018.

[10] Goodfellow IJ, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y. Generative adversarial networks. arXiv:1406.2661, 2014.

[11] Mohseni S, Pitale M, Yadawa J, Wang ZY. Self-supervised learning for generalizable out-of-distribution detection. Proc. of the AAAI Conf. on Artificial Intelligence, 2020, 34(4): 5216–5223. [doi: 10.1609/aaai.v34i04.5966]

[12] Chen JF, Li YX, Wu X, Liang YY, Jha S. Robust out-of-distribution detection for neural networks. arXiv:2003.09711, 2021.

[13] Augustin M, Meinke A, Hein M. Adversarial robustness on in- and out-distribution improves explainability. In: Proc. of the 16th European Conf. on Computer Vision. Glasgow: Springer, 2020. 228–245. [doi: 10.1007/978-3-030-58574-7_14]

[14] Sehwag V, Bhagoji AN, Song LW, Sitawarin C, Cullina D, Chiang M, Mittal P. Analyzing the robustness of open-world machine learning. In: Proc. of the 12th ACM Workshop on Artificial Intelligence and Security. London: ACM, 2019. 105–116. [doi: 10.1145/3338501.3357372]

[15] Hein M, Andriushchenko M, Bitterwolf J. Why ReLU networks yield high-confidence predictions far away from the training data and how to mitigate the problem. In: Proc. of the 2019 IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR). Long Beach: IEEE, 2019. 41–50. [doi: 10.1109/CVPR.2019.00013]

[16] Chen JF, Li YX, Wu X, Liang YY, Jha S. Atom: Robustifying out-of-distribution detection using outlier mining. In: Proc. of the 2021 European Conf. on Machine Learning and Knowledge Discovery in Databases. Bilbao: Springer, 2021. 430–445. [doi: 10.1007/978-3-030-

86523-8_26]

[17] Szegedy C, Zaremba W, Sutskever I, Bruna J, Erhan D, Goodfellow I, Fergus R. Intriguing properties of neural networks. arXiv: 1312.6199, 2014.

[18] Goodfellow IJ, Shlens J, Szegedy C. Explaining and harnessing adversarial examples. arXiv:1412.6572, 2015.

[19] Xu H, Ma Y, Liu HC, Deb D, Liu H, Tang JL, Jain AK. Adversarial attacks and defenses in images, graphs and text: A review. Int'l Journal of Automation and Computing, 2020, 17(2): 151–178. [doi: 10.1007/s11633-019-1211-x]

[20] Madry A, Makelov A, Schmidt L, Tsipras D, Vladu A. Towards deep learning models resistant to adversarial attacks. arXiv:1706.06083, 2019.

[21] Zhang HY, Yu YD, Jiao JT, Xing E, El Ghaoui L, Jordan MI. Theoretically principled trade-off between robustness and accuracy. In: Proc. of the 36th Int'l Conf. on Machine Learning. Long Beach: PMLR, 2019. 7472–7482.

[22] Tsipras D, Santurkar S, Engstrom L, Turner A, Madry A. Robustness may be at odds with accuracy. arXiv:1805.12152, 2019.

[23] Bai Y, Feng Y, Wang YS, Dai T, Xia ST, Jiang Y. Hilbert-based generative defense for adversarial examples. In: Proc. of the 2019 IEEE/CVF Int'l Conf. on Computer Vision (ICCV). Seoul: IEEE, 2019. 4783–4792. [doi: 10.1109/ICCV.2019.00488]

[24] Ma XJ, Li B, Wang YS, Erfani SM, Wijewickrema S, Schoenebeck G, Song D, Houle ME, Bailey J. Characterizing adversarial subspaces using local intrinsic dimensionality. arXiv:1801.02613, 2018.

[25] Xu WL, Evans D, Qi YJ. Feature squeezing: Detecting adversarial examples in deep neural networks. arXiv:1704.01155, 2017.

[26] Grosse K, Manoharan P, Papernot N, Backes M, McDaniel P. On the (statistical) detection of adversarial examples. arXiv:1702.06280, 2017.

[27] Carlini N, Wagner D. Adversarial examples are not easily detected: Bypassing ten detection methods. In: Proc. of the 10th ACM Workshop on Artificial Intelligence and Security. Dallas: ACM, 2017. 3–14. [doi: 10.1145/3128572.3140444]

[28] Croce F, Hein M. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In: Proc. of the 37th Int'l Conf. on Machine Learning. Virtual Event: JMLR.org, 2020. 2206–2216.

[29] Tramèr F, Kurakin A, Papernot N, Goodfellow I, Boneh D, McDaniel P. Ensemble adversarial training: Attacks and defenses. arXiv:1705.07204, 2020.

[30] Moosavi-Dezfooli SM, Fawzi A, Fawzi O, Frossard P. Universal adversarial perturbations. In: Proc. of the 2017 IEEE Conf. on Computer Vision and Pattern Recognition. Honolulu: IEEE, 2017. 86–94. [doi: 10.1109/CVPR.2017.17]

[31] Moosavi-Dezfooli SM, Fawzi A, Frossard P. DeepFool: A simple and accurate method to fool deep neural networks. In: Proc. of the 2016 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR). Las Vegas: IEEE, 2016. 2574–2582. [doi: 10.1109/CVPR.2016.282]

[32] Kurakin A, Goodfellow I, Bengio S. Adversarial examples in the physical world. arXiv:1607.02533, 2017.

[33] Papernot N, McDaniel P, Goodfellow I, Jha S, Celik ZB, Swami A. Practical black-box attacks against machine learning. In: Proc. of the 2017 ACM on Asia Conf. on Computer and Communications Security. Abu Dhabi: ACM, 2017. 506–519. [doi: 10.1145/3052973.3053009]

[34] Papernot N, McDaniel P, Jha S, Fredrikson M, Celik ZB, Swami A. The limitations of deep learning in adversarial settings. In: Proc. of the 2016 IEEE European Symp. on Security and Privacy (EuroS&P). Saarbruecken: IEEE, 2016. 372–387. [doi: 10.1109/EuroSP.2016.36]

[35] Pan WW, Wang XY, Song ML, Chen C. Survey on generating adversarial examples. Ruan Jian Xue Bao/Journal of Software, 2020, 31(1): 67–81 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/5884.htm [doi: 10.13328/j.cnki.jos.005884]

[36] Tramèr F, Carlini N, Brendel W, Madry A. On adaptive attacks to adversarial example defenses. In: Proc. of the 34th Int'l Conf. on Neural Information Processing Systems. Vancouver: Curran Associates Inc., 2020. 1633–1645.

[37] Kurakin A, Goodfellow I, Bengio S. Adversarial machine learning at scale. arXiv:1611.01236, 2017.

[38] Athalye A, Carlini N, Wagner D. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In: Proc. of the 35th Int'l Conf. on Machine Learning. Stockholm: PMLR, 2018. 274–283.

[39] Papernot N, McDaniel P, Wu X, Jha S, Swami A. Distillation as a defense to adversarial perturbations against deep neural networks. In: Proc. of the 2016 IEEE Symp. on Security and Privacy (SP). San Jose: IEEE, 2016. 582–597. [doi: 10.1109/SP.2016.41]

[40] Carlini N, Wagner D. Towards evaluating the robustness of neural networks. In: Proc. of the 2017 IEEE Symp. on Security and Privacy (SP). San Jose: IEEE, 2017. 39–57. [doi: 10.1109/SP.2017.49]

[41] Gowal S, Uesato J, Qin CL, Huang PS, Mann T, Kohli P. An alternative surrogate loss for PGD-based adversarial testing. arXiv:1910.09338, 2019.

[42] Gowal S, Qin CL, Uesato J, Mann T, Kohli P. Uncovering the limits of adversarial training against norm-bounded adversarial examples. arXiv:2010.03593, 2021.

[43] Krizhevsky A. Learning multiple layers of features from tiny images. 2009. http://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf
[44] Zagoruyko S, Komodakis N. Wide residual networks. arXiv:1605.07146, 2017.
[45] Netzer Y, Wang T, Coates A, Bissacco A, Wu B, Ng AY. Reading digits in natural images with unsupervised feature learning. In: Proc. of the 25th Conf. on Neural Information Processing Systems. 2011.
[46] Torralba A, Fergus R, Freeman WT. 80 million tiny images: A large data set for nonparametric object and scene recognition. IEEE Trans. on Pattern Analysis and Machine Intelligence, 2008, 30(11): 1958–1970. [doi: 10.1109/TPAMI.2008.128]
[47] Zhou BL, Lapedriza A, Khosla A, Oliva A, Torralba A. Places: A 10 million image database for scene recognition. IEEE Trans. on Pattern Analysis and Machine Intelligence, 2018, 40(6): 1452–1464. [doi: 10.1109/TPAMI.2017.2723009]
[48] Cimpoi M, Maji S, Kokkinos I, Mohamed S, Vedaldi A. Describing textures in the wild. In: Proc. of the 2014 IEEE Conf. on Computer Vision and Pattern Recognition. Columbus: IEEE, 2014. 3606–3613. [doi: 10.1109/CVPR.2014.461]
[49] Xu PM, Ehinger KA, Zhang YD, Finkelstein A, Kulkarni SR, Xiao JX. TurkerGaze: Crowdsourcing saliency with webcam based eye tracking. arXiv:1504.06755, 2015.
[50] Yu F, Seff A, Zhang YD, Song SR, Funkhouser T, Xiao JX. LSUN: Construction of a large-scale image dataset using deep learning with humans in the loop. arXiv:1506.03365, 2016.

附中文参考文献:

[35] 潘文雯, 王新宇, 宋明黎, 陈纯. 对抗样本生成技术综述. 软件学报, 2020, 31(1): 67–81. http://www.jos.org.cn/1000-9825/5884.htm [doi: 10.13328/j.cnki.jos.005884]

周志阳(1990－), 男, 博士生, CCF 学生会员, 主要研究领域为可信人工智能, 大数据系统.

王帅(1982－), 男, 博士, 高级工程师, 主要研究领域为大数据分析处理, 人工智能, 系统集成.

窦文生(1984－), 男, 博士, 研究员, CCF 专业会员, 主要研究领域为程序分析, 软件工程.

刘杰(1982－), 男, 博士, 副研究员, CCF 专业会员, 主要研究领域为大数据, 分布式系统, 软件工程.

李硕(1997－), 女, 博士生, CCF 学生会员, 主要研究领域为智能化软件工程.

叶丹(1971－), 女, 博士, 研究员, 博士生导师, CCF 高级会员, 主要研究领域为网络分布式系统, 软件工程.

亢良伊(1993－), 女, 博士, 主要研究领域为深度学习及其优化, 自然语言处理.