# 一种基于知识蒸馏的神经网络鲁棒性迁移方法

张 维, 易 平

上海交通大学网络空间安全学院 上海 中国 200240

**摘要** 近几年来, 深度神经网络在多个领域展现了非常强大的应用能力, 但是研究者们发现, 通过在输入上添加难以察觉的扰动, 可以改变神经网络的输出决策, 这类样本被称为对抗样本。目前防御对抗样本, 最常见的方法是对抗训练, 但是对抗训练有着非常高的训练代价。我们提出了一种知识蒸馏的鲁棒性迁移方案(Robust-KD), 结合特征图与雅克比矩阵约束, 通过从鲁棒的网络中迁移鲁棒性特征, 以比较低的训练代价, 取得较强的白盒对抗防御能力。提出的算法在 Cifar10、Cifar100 与 ImageNet 数据集上进行了大量的实验, 实验表明了我们方案的有效性, 即使在非常强大的白盒对抗攻击下, 我们的模型依然拥有不错的分类准确率。

**关键词** 对抗样本; 模型鲁棒性; 迁移学习; 知识蒸馏

**中图法分类号** TP18 DOI 号 10.19363/J.cnki.cn10-1380/tn.2021.07.04

# A Robust Transfer Method of Neural Network based on Knowledge Distillation

ZHANG Wei, YI Ping

School of Cyber Science and Engineering, Shanghai Jiao Tong University, Shanghai 200240, China

**Abstract** In recent years, neural networks have shown very powerful performance in many fields, but researchers have found that by adding imperceptible interference to the input, neural network decisions can be changed. Such samples are called adversarial samples. At present, the most common method for defending adversarial examples is adversarial training, but the training cost of adversarial training is very high. We propose a knowledge purification scheme (Robust-KD) combining feature maps and Jacobian matrix constraints. By migrating robust features from a robust network, we can obtain considerable white box defense capabilities at relatively low training costs. We have conducted a lot of experiments on the Cifar10, Cifar100 and ImageNet datasets. Experiments have proved the effectiveness of the scheme. Even under a very powerful white box attack, our model still has good classification accuracy.

**Key words** adversarial examples; model robustness; transfer learning; knowledge distillation

## 1 引言

深度学习模型在众多计算机视觉领域[1-3], 取得非常出色的效果。人工智能算法已经渗透到了我们的日常生活中的方方面面, 例如人脸识别系统, 城市大脑等等。但是, 深度学习算法极其容易被对抗样本攻破[4,36] , 通过在图像输入中, 添加极其微小的扰动, 就可以改变深度学习模型的决策。在一些对于安全性要求非常高的场景下, 例如自动驾驶的感知, 对抗样本的危害显得尤为明显, Eykholt 等人[5]提出了一种通用的攻击框架, 通过给路标添加扰动, 使得机器将“右转”标识识别为“停止”标识, 将“停止”标识识别为“限速”标识等等, 这类标识在不同的拍摄角度下, 都可以完成对相机感知系统的攻击, 如下图 1 所示。对抗样本在自动驾驶场景下, 会带来


**图 1 交通场景下的对抗样本**

**Figure 1 Adversarial examples in traffic scenarios**

**通讯作者:** 易平, 博士, 副教授, Email: yiping@sjtu.edu.cn。

本课题得到国家重点研发计划(No. 2019YFB1405000)资助。

收稿日期: 2020-10-14; 修改日期: 2020-12-08; 定稿日期: 2021-06-24

难以估量的损失, 因此, 防御对抗样本对于深度学习算法的应用具有重大价值。

因为对抗样本在深度学习领域潜在的安全风险, 近些年来, 对抗样本一直是学术界的研究重点, 涌现了大批关于攻击与防御的工作。目前最有效的防御对抗样本的方法是对抗训练[6], 对抗训练的思想非常简单但也非常有效, 即在训练中将生成的对抗样本加入到训练集中, 从而对网络进行迭代优化, 将 min-max 的优化结合到训练中。但是对抗训练的劣势也非常明显, 生成强大的对抗样本需要多次迭代, 即神经网络需要多次的反向传播, 导致训练的代价大大增加, 从训练时间长来看, 对抗训练一般是正常训练的数倍, 一定程度上限制了对抗训练的应用。

为了减少对抗训练的代价, 近期出现了用迁移学习的方法提升模型鲁棒性的工作, 即将神经网络的鲁棒性从一个模型中迁移到其他的任务上[7]。一种方法将神经网络的参数作为媒介迁移鲁棒性, 将模型的基础层固定, 仅在新的任务上微调最后几层的参数, 微调后的网络仍然保留了非常不错的鲁棒性; 另一种思路利用知识蒸馏[7-8]迁移模型的鲁棒性, 将鲁棒模型作为教师网络, 利用 KL 散度约束教师网络与学生网络之间的表征层特征, 知识蒸馏使用的是模型的表征层特征作为迁移媒介。这两种思路, 均取得了不错的白盒防御效果。

我们提出了一种鲁棒性迁移的方法(Robust-KD), 不通过对抗训练, 就可以获得具有很强白盒防御能力的模型。受注意力迁移[9]与鲁棒模型性质[10]启发, 我们利用特征更加丰富的特征图作为知识蒸馏的媒介迁移鲁棒性, 与表征层不同, 浅层的特征图编码了低层次的特征, 深层的特征图编码了高层次的语义特征, 特征图包含的信息量更加丰富; 同时, 我们在使用特征图作为约束的基础上, 还加入了 Jacobian 矩阵的约束, Jacobian 矩阵很好地反映了深度网络的梯度特征, 我们通过计算特征图与输入层之间的 Jacobian 矩阵, 并且使用了一种高效的方法, 将 Jacobian 矩阵作为约束, 融合到我们知识蒸馏模型的求解框架中。

迁移学习[11]的成功是计算机视觉的里程碑之一, 在较大数据集上(例如 ImageNet[12])学到的特征, 可以很好地迁移到其他任务上, 预训练方法已经成为了深度学习领域最常见的一种方法。相应地, 我们在 ImageNet 数据集上进行对抗训练, 得到鲁棒的神经网络, 该鲁棒网络很好地编码了通用的鲁棒性特征。在一些小的任务上(例如 Cifar10)进行知识蒸馏, 以较小的代价, 就可以获取很强的对抗样本防御能力。

通过在 ImageNet、Cifar100、Cifar10 上的实验结果, 显示了本文提出的知识蒸馏方案的有效性, 特征图与 Jacobian 矩阵的约束, 很好的增强了模型的白盒防御能力。在扰动较小时, 学生网络的防御能力甚至已经接近对抗训练, 而且在干净样本的准确率上, 本文提出的训练方法也非常可观, 相较正常训练, 并没有损失很多。

总之, 我们的贡献可以概括为以下三点:

1) 提出了一种鲁棒性迁移的方法, 可以以较低的代价(非对抗训练), 获得不错的对抗样本防御能力。

2) 验证了特征图与 Jacobian 矩阵约束, 在提升模型鲁棒性上的有效性。

3) 做了大量的实验, 为鲁棒性迁移提供了一个很好的基线。

## 2 背景与相关工作

我们回顾下对抗样本模型鲁棒性的一些概念, 并介绍本文用到的基于梯度的攻击算法, 目前的对抗样本防御算法以及知识蒸馏等等。

### 2.1 模型鲁棒性

我们假定分类模型为 $f(x;\theta): x \to R^k$, 将输入图片 $x$ 映射为 $k$ 类别的概率分布, 神经网络的参数被定义为 $\theta$。假设神经网络的训练集为 $D$, 那么该模型的训练目标可以定义为 $\min_\theta E_{(x,y)\in D} CE(x,y)$, 其中 $y \in R^k$ 是训练集中的类别标签, 以 one-hot 编码方式编码, $CE(x,y)$ 是分类常用的交叉熵损失函数。

$$CE(x,y) = E_{(x,y)\in D}\left[-y^T \log\left(f(x;\theta)\right)\right] \tag{1}$$

在干净的测试样本上, 通过该训练目标得到的深度学习模型, 可以获得非常不错的效果, 但是在对抗样本上, 该模型的准确率下降非常多。我们在这里给出鲁棒模型的定义, 一个鲁棒的模型应该满足以下条件

$$f(x;\theta) = f(x+\sigma;\theta) \tag{2}$$

其中, $\sigma \in N_\sigma(x)$ 是添加到输入图片上的扰动, 为了保证 $\sigma$ 的扰动不过于明显, $\sigma$ 同时还需要满足范数的约束条件 $\left|\left|\sigma\right|\right|_p \leqslant \epsilon$, 通常情况下, $p=2$ 或者 $p=\infty$ 是目前研究的重点, 在我们的实验中, 我们针对这两种范数约束的对抗样本, 均进行了大量的实验。

在鲁棒模型的定义下, 鲁棒模型的决策并不会因为对抗扰动的加入而改变, 因此鲁棒模型在对抗样本上依然会保留可观的准确率, 在后续的实验部

分, 我们也会通过模型在不同扰动程度下的对抗样本上的准确率, 表明我们方法的有效性。

## 2.2 基于梯度的攻击算法

生成对抗样本可以看成一个带有限制的优化问题, 优化的目标方程为

$$\text{argmax}_{x^{adv}} L(x^{adv}, y), st. \left|\left|x^{adv} - x^{clean}\right|\right|_p \leqslant \epsilon \quad (3)$$

为了求解这个优化问题, 在白盒攻击的场景下, 可以将梯度反传到输入层, 求解噪声使得交叉熵损失函数变大。一些工作已经被提出来求解上述的目标方程, 这里我们给出一些简单的介绍。

### 2.2.1 快速梯度标志攻击

快速梯度标志攻击(Fast Gradient Sign Method, FGSM)[13], 通过反向传播, 将损失函数反传到输入层, 并且通过梯度方向更新输入图片生成对抗样本, 从而达到欺骗分类器的目的, 其生成对抗样本的方法如下

$$x^{adv} = x^{clean} + \epsilon sign\left(\nabla_x L(x^{clean}, y)\right) \quad (4)$$

其中 $\nabla_x L$ 是反传回输入层的梯度, $sign(.)$ 是符号函数, $\epsilon$ 一般用来调整 FGSM 攻击的扰动程度, $\epsilon$ 越大, 生成的对抗样本攻击能力越强。

FGSM 算法较为简单, 可以快速生成大量的对抗样本, 但是生成的对抗样本大多攻击能力比较一般。

### 2.2.2 映射梯度下降法

映射梯度下降法(Projected Gradient Descend, PGD)[6], 是 FGSM 算法的改进, 可以看成是多步迭代版本的 FGSM, PGD 为了保证扰动满足 $L_2$ 或者 $L_\infty$ 的限制, 在迭代的过程中引入了映射的过程, 将扰动映射回 $L_2$ 或者 $L_\infty$ 的球中, 其迭代过程如下

$$x_0^{adv} = x^{clean} \quad (5)$$

$$x_{t+1}^{adv} = proj_x\left(x_t^{adv} + \epsilon sign\left(\nabla_x L(x_t^{adv}, y)\right)\right) \quad (6)$$

与 FGSM 相比, PGD 算法攻击强度更强, 但是生成对抗样本的耗时也是 FGSM 的数倍, 大多数情况下, PGD 迭代次数越多, 生成的对抗样本攻击能力越强。

### 2.2.3 动量迭代梯度标志攻击

动量迭代梯度标志攻击(Momentum Iterative Fast Gradient Sign Method, MI-FGSM)[25], 是由 Dong 提出的一种增强攻击迁移性的方法。过去的迭代式的攻击方法(PGD), 黑盒攻击效果很差, 迭代得到的解容易陷入局部最优点。Dong 将优化器中常用的动量(Momentum)引入到对抗样本的生成中, 其中梯度方法的积累公式如下

$$g_{t+1} = \mu \cdot g_t + \frac{\nabla_x L(x_t^{adv}, y)}{|\nabla_x L(x_t^{adv}, y)|_1} \quad (7)$$

$$x_{t+1}^{adv} = proj_x\left(x_t^{adv} + \epsilon sign(g_{t+1})\right) \quad (8)$$

与 PGD 攻击算法的区别在于, 更新梯度的过程中, 通过 $\mu$ 控制累积梯度的更新速度, 从而很好的缓解了过拟合现象, 大大提升了算法的黑盒迁移效果。

### 2.2.4 Carlini-Wagner 攻击

Carlini-Wagner 攻击(CW 攻击)[26], 是一种基于优化的对抗样本生成方法, 与 PGD 算法不同, PGD 算法通过投影的方法限制扰动的幅度, 是非常粗粒度的; 而 CW 则是在优化的目标函数的中, 加入了扰动的范数, 在保证攻击效果的同时, 使求解得到的扰动尽可能小。CW 函数的优化目标为

$$\min_{\delta \in R^n}(|\delta|_p - L(x_t^{adv} + \delta, y)) \quad (9)$$

其中 $|\cdot|_p$ 为扰动的范数, 该目标函数保证了 CW 攻击能够以较小的扰动实现强有力的攻击。虽然相比 PGD 攻击算法, CW 攻击可以构造出扰动更小的对抗样本, 但是 CW 攻击的复杂度也比 PGD 攻击算法要高得多。

## 2.3 现有的防御算法

已有的防御方法可以分为两个主要流派, 一种是使用对抗训练的防御方法, 即在训练中实时生成对抗样本, 并将生成的对抗样本作为训练集反哺给模型; 另一个是非对抗训练的方法, 会更加关注于深度模型的一些性质。

### 2.3.1 对抗训练防御

对抗训练(Adversarial Training, AT)[6], 是 Madry 提出的一种对抗样本防御的方法, 也被称为 Madry Defense, 是目前防御对抗样本最为有效的一种方式。Madry 在[6]的工作中, 将防御对抗样本定义为一个 min-max 优化的问题, 生成对抗样本的目标是为了增大损失函数, 使深度学习模型决策错误; 防御对抗样本又是为了减少这类样本的损失函数, 这样就形成了一种对抗, Madry 给出了一种求解方式

$$L(x, y) = E_{(x,y)\in D}\left[\max_{\sigma \in N_\epsilon(x)} L(x + \sigma, y)\right] \quad (10)$$

其中 $\sigma$ 为对抗扰动, 通过 PGD 算法求解得到。对抗训练在防御对抗样本上效果非常好, 几乎对于所有基于一阶导数的攻击, 都有着非常不错的防御效果。考虑到 PGD 攻击算法的复杂度, 对抗训练的缺点同样非常明显, 对抗训练的代价是正常训练的数倍, 这也限制了对抗训练算法的应用。

后续也有非常多的工作改进对抗训练, Zhang 等人[14]使用对抗样本与干净样本特征之间的距离作为

目标函数, 生成了更加高效的对抗样本, 提升了对抗训练的效率。Zhang 等人[15]从平滑模型决策边界出发, 构造正则项, 通关该正则项约束对抗样本与干净样本之间的决策偏差, 提升了对抗训练得到的模型的泛化性。Qin[27]认为对抗的耗时是由神经网络的非线性导致的, Qin 提出了一种正则化的方法, 约束真实损失函数与估计的线性损失函数之间的差, 从而惩罚梯度混淆, 提升了对抗训练的效果。

### 2.3.2 非对抗训练防御

近几年也有一些不使用对抗训练的防御策略, Ross 等人[16]分析鲁棒模型的性质, 发现对抗训练得到的模型在输入层梯度上, 与普通模型呈现出不同的模式, 如下图 2 所示, Ross 利用了这一特性, 通关两次反向传播的方法[28]约束输入层梯度的 Frobenius norm, 取得了不错的防御效果。

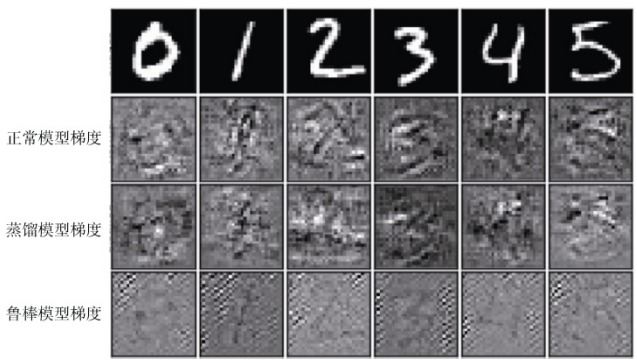


**图 2 模型输入层梯度可视化**

**Figure 2 The visualization of input layers' gradient**

与 Ross 的策略类似, Hoffman 等人[17]利用 Jacobian 矩阵作为神经网络的正则项约束网络求解, Hoffman 从泰勒展开出发, 证明了约束 Jacobian 矩阵的 $L_2$ 范数可以使模型决策边界更加平滑, Hoffman 给出了一个随机投影的策略, 很好地将 Jacobian 矩阵作为损失函数, 加入到了神经网络的求解过程中。

Chan 等[29]很好地运用了 Ross 提出的性质, 他认为一个鲁棒模型的梯度图像应该与原始图像存在对应关系, 所有 Chan 构造了一种正则损失函数, 使得模型梯度图像与原始图片相对应, 大大增强了模型鲁棒性。还有一些工作[30-31]尝试从模型性质上, 寻找对抗样本生成的边界, 这类防御首先从理论上证明了边界的存在, 再优化这个边界, 使得对抗样本更难生成, 从而达到增强模型鲁棒性的目的。

我们的工作首次尝试将 Jacobian 矩阵作为知识蒸馏的载体, 在提升模型鲁棒性上取得了不错的效果。

### 2.3.3 对抗样本检测

上述两节主要关注的是模型鲁棒性, 即模型对于对抗样本具有容忍性, 可以将对抗样本分类正确, 目前还有一种防御对抗样本的思路, 即将异常的对抗样本检测出。对抗样本检测算法关注的是如何衡量正常样本的分布, 从大量的数据中找出异常点, 使用统计学的方法或者基于神经网络的方法挖掘异常特征。

Ma 等人[32]提出了局部本征维度(Local Intrinsic Dimensionality, LID)的方法, 检测对抗样本是否存在, LID 通过 K-Means 算法[33]刻画正常样本的分布, 将干净样本聚类得到 $N$ 个中心, 越远离聚类中心的点, 是对抗样本的概率越高。Meng 等人[34]提出了一种基于重构误差检测对抗样本的方法, Meng 训练了一个自编码器来实现样本的编码与解码, 通过自编码器学习干净样本的分布, 因为对抗样本并未出现在自编码器的训练数据中, 对抗样本编码解码后得到的图像与输入图像差别大于干净样本编解码, 通过这种方法实现对抗样本的检测。Xu 等人[35]提出了一种基于特征压缩(Feature Squeezing)的对抗样本检测方案, 该方法将原样本(0~255 bit)压缩到新的像素空间(0~8 bit), 像素空间的压缩, 可以减少对抗样本中微小扰动所带来的误差, 将压缩后的图像结果与原样本结果比对, 从而分辨出原样本是否为对抗样本。

## 2.4 知识蒸馏

知识蒸馏这个概念最早由Hinton等[8]提出, 一般认为神经网络训练中编码了很强的先验知识, 这些知识在不同的任务之间具有一定的迁移能力, Hinton 设计了一种训练框架, 该框架中含有两个神经网络, 一个被称为教师网络, 另一个为学生网络。Hinton先训练好教师网络, 后将教师网络的输出作为学生网络的监督, 这种范式常被用来训练轻量级网络, 使用大网络的输入监督小网络, 小网络可以保留接近大网络的性能。

Shafahi 等人[7]首次用知识蒸馏的方法来训练鲁棒模型, Shafahai 使用在 Tiny-ImageNet[18]上对抗训练得到的模型作为教师模型, 使用鲁棒模型的输出来监督新网络的学习, 取得了非常不错的防御效果。这种模式很好地将迁移学习的思想融入到模型鲁棒性的增强上来, 以教师网络输出与学生网络输出之间的 KL 散度约束迁移鲁棒性。

但是知识蒸馏是为了模型在干净样本上的准确率的提升而设计的, 在鲁棒性迁移上, 模型的输入是干净样本而不是对抗样本, 我们猜测基于 KL 散度的知识蒸馏并非是最优的选择, 在本文中我们简单地探讨了在鲁棒性迁移场景下, 知识蒸馏方法的改进。

## 3 神经网络鲁棒性迁移

我们的工作解决的是鲁棒性迁移的问题, 即如何将鲁棒的教师网络的特征迁移到正在训练的学生网络中。我们的工作分为三部分叙述, 第一部分是方法的概述; 第二部分是基于特征图的鲁棒性迁移; 第三部分是基于 Jacobian 矩阵的鲁棒性迁移。

### 3.1 总览

首先简单介绍下我们的方法, 教师网络在较大的数据集(例如 ImageNet)上通过分类任务训练得到。我们期望该鲁棒网络的特征是可以迁移到多个任务上的, 相较于 Hinton 的早期版本的知识蒸馏, 本文期望构建一个更加通用的方法。

受 Li 等人[19]与 Shafahi 等人[7]工作的启发, 我们设计了一种蒸馏的框架, 学生网络与教师网络在结构上保持一致, 因为目标任务类别会与 ImageNet 不一致, 所以以神经网络中的特征图作为迁移鲁棒性的媒介, 而非基于 logits, 方法的整体框架见下图 3。

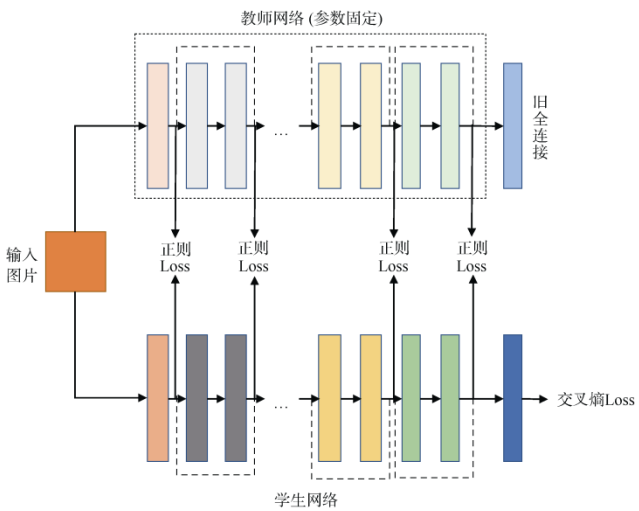


**图 3 鲁棒性迁移框架**

**Figure 3 The framework of robustness transfer**

假设教师网络为 $f_{teacher}$, 学生网络为 $f_{student}$, 教师网络与学生网络的特征图分为 $z_i^{teacher}$, $z_i^{student}$, $z_i^{teacher}$ 表示教师网络前向时输出的第 $i$ 个特征图(feature map), 特征图满足

$$z_0^{teacher}, z_1^{teacher}, z_2^{teacher} \ldots = f_{teacher}(x) \quad (11)$$

$$z_0^{student}, z_1^{student}, z_2^{student} \ldots = f_{student}(x) \quad (12)$$

Shafahi 等人[7]的知识蒸馏, 只在最后一个特征层上约束(Global Average Pooling, GAP), 池化的操作损失了大量的空间信息, 其损失函数为

$$L(x,y) = CE(x,y) + \alpha Reg\left(GAP\left(z_{-1}^{student}\right), GAP\left(z_{-1}^{teacher}\right)\right) \quad (13)$$

其中 $CE(.)$ 为分类的交叉熵损失, $Reg$ 为正则损失, 该工作使用的是 $L_2$ loss, $\alpha$ 为调整损失函数权重的超参。我们的工作将约束扩展到了更多的层上(以 ResNet[22]为例, 约束被添加在每个 res-block 后), 并且摒弃了 Global Average Pooling, 保留了最后一层特征的空间信息, 框架损失函数为

$$L(x,y) = CE(x,y) + \sum_{i=0}^{n} \alpha Reg\left(z_i^{student}, z_i^{teacher}\right) \quad (14)$$

在下面两节中, 我们将详细介绍正则项的设计。

### 3.2 基于特征图的鲁棒性迁移

在本部分, 我们将介绍基于特征图约束的方法, 以及约束方式的选择。在衡量两个分布时, 通常选用的指标有 KL 散度[20]、$L_2$ 距离、Cosine 距离[21]等。

大多数知识蒸馏工作一般也使用的是 $L_2$ 距离作为衡量教师网络特征与学生网络特征的指标。但是在实验中发现, 因为教师网络训练的数据集与当前训练的数据集, 会存在较大的分布上的差异, 这种差异会导致学生网络与教师网络输出的特征图, 在量级上存在些许差异, 而 $L_2$ 距离对于量级的变化极其敏感。

所以在训练中, 我们引入了 Cosine 距离作为衡量教师网络特征图与学生网络特征图之间差异的指标, Cosine 距离的定义如下

$$Cos(x,y) = \frac{x}{\|x\|_2} \cdot \frac{y}{\|y\|_2} \quad (15)$$

其中 $\left|\|x\|\right|_2$ 是矩阵的 $L_2$ 范数, 当 $x$ 与 $y$ 越接近, $Cos(x,y)$的值为 1, 当 $x$ 与 $y$ 越不接近, $Cos(x,y)$的值为–1, 相较于 $L_2$ 距离, Cosine 距离的优势在于它对输入的向量都会做归一化, 解决了不同域分布不同带来的参数 scale 的影响, 在我们的实验中, Cosine 距离效果也优于 $L_2$ 距离。

在具体实现时, 教师网络的参数是固定的, 且不进行反向传播的, 对于目标数据集的图片输入, 仅保留教师网络的每一层的特征图, 对于学生网络, 训练的损失函数为交叉熵损失与 Cosine 正则损失之和

$$L_{fm} = \sum_{i=0}^{n} \left[1 - cos\left(z_i^{student}, z_i^{teacher}\right)\right] \quad (16)$$

通过这一损失函数, 可以在不损失过多准确率的前提下, 尽可能保留鲁棒教师网络的各层特征表达。为了提升效果, 学生网络使用教师网络作为预训练模型, 在教师网络的参数基础上微调, 后面所有的实验也都建立在使用预训练模型的基础上。

### 3.3 基于 Jacobian 矩阵的鲁棒性迁移

对于分类模型, 以图片为输入, 以类别概率为

输出, $z=f(x;\theta)\in R^k$, 一个鲁棒的模型对于微小的噪声应该是不敏感的, 满足 $f(x;\theta)=f(x+\sigma;\theta)$, 假设噪声为 $\sigma$, 我们对 $f(x+\sigma;\theta)$ 进行泰勒展开

$$f(x+\sigma;\theta)=f(x;\theta)+\sum_{i=0}^{I}\sigma_i\frac{\partial f}{\partial x_i}(x;\theta)+O(\sigma^2)$$
$$=f(x;\theta)+\sum_{i=0}^{I}J_i(x;\theta)+O(\sigma^2) \quad (17)$$

$J_i(x;\theta)$ 为输入 $x_i$ 对应的 Jacobian 矩阵, 通过泰勒展开我们可以发现, 模型对于噪声的响应, 与 Jacobian 矩阵存在着非常高的关联性。对于一个鲁棒的模型, 其满足 $f(x;\theta)=f(x+\sigma;\theta)$, 即 $\sum_{i=0}^{I}J_i(x;\theta)+O(\sigma^2)=0$, 可以发现模型是否鲁棒与神经网络的 Jacobian 矩阵具有非常强的相关性, 对抗训练后的鲁棒性网络对于对抗样本是有一定的鲁棒性的, 在鲁棒性知识蒸馏中, 我们期望学生网络获得鲁棒的教师网络的 Jacobian 矩阵性质, 所以在此处我们将尝试以 Jacobian 矩阵作为鲁棒性迁移的媒介, 尝试从鲁棒教师网络中迁移鲁棒性。

与 3.2 一致, 我们期望在神经网络的多层中间层中添加 Jacobian 约束, 但是中间层的 Jacobian 矩阵维度极高, 用来直接约束神经网络的求解不现实, 所以需要设计一种降维的方法。受 Hoffman 等人[17]工作的启发, 这里我们引入了一种随机投影的方法, 通过多次投影, 去近似完整的 Jacobian 矩阵, 从而大大降低了运算的复杂度。假定图片输入的维度为 $I$, 中间特征层的维度为 $O$, 那么中间特征层相对图片输入的 Jacobian 矩阵为 $J\in R^{I\times 0}$ 满足

$$J(z,x)=\begin{bmatrix}\frac{\partial z_0}{\partial x_0} & \cdots & \frac{\partial z_{O-1}}{\partial x_0}\\ \vdots & \ddots & \vdots\\ \frac{\partial z_0}{\partial x_{I-1}} & \cdots & \frac{\partial z_{O-1}}{\partial x_{I-1}}\end{bmatrix} \quad (18)$$

在训练时, 每一次求 Jacobian 矩阵时, 我们都会随机生成一个 $L_2$ 范数为 1 的 mask, 这里记为 $M\in R^O$, 通过 mask 与特征图相乘, 求得中间特征图在 mask 方向投影后的长度, 通过这种方法, 将特征图转化为标量, 大大减少了计算复杂度, 简化后的 Jacobian 矩阵为

$$J(z,x,M)=\begin{bmatrix}\frac{\partial M^T z}{\partial x_0}\\ \vdots\\ \frac{\partial M^T z}{\partial x_{I-1}}\end{bmatrix} \quad (19)$$

基于 Jacobian 矩阵的训练损失函数可以被定义为

$$L_{jacobian}=\sum_{i=0}^{n}\sum_{j=0}^{n_{proj}}\left[J(z^i_{teacher},x,M_j)-J(z^i_{student},x,M_j)\right]^2 \quad (20)$$

其中 $n_{proj}$ 为投影映射的次数, $n_{proj}$ 越大计算复杂度越高, 但是能够更好地逼近特征图与输入之间的 Jacobian 矩阵。算法的具体流程见算法 1。

算法 1 基于 Jacobian 矩阵的鲁棒性迁移

**输入**: 目标数据集 $\boldsymbol{D}$, 教师网络 $\boldsymbol{f_{teacher}}$, 学生网络 $\boldsymbol{f_{student}}$ 学习率 $\boldsymbol{\beta}$, 随机投影次数 $\boldsymbol{n_{proj}}$

**输出**: 鲁棒的学生模型 $\boldsymbol{f_{student}}$

**begin**
**for** training Iteration **do**
从 $\boldsymbol{D}$ 中随机采样 $(x, y)$
$\boldsymbol{z_0^{teacher}, z_1^{teacher}} \ldots \leftarrow \boldsymbol{f_{teacher}(x)}$ ▷ 前向计算教师网络特征图
$\boldsymbol{z_0^{student}, z_1^{student}} \ldots \leftarrow \boldsymbol{f_{student}(x)}$ ▷ 前向计算学生网络特征图
for $\boldsymbol{j \in [0,1,\ldots,n_{proj}-1]}$ do
生成随机 mask $\boldsymbol{M_j}$
$\boldsymbol{J^i_{student}\leftarrow J(z^i_{teacher}, x, M_j)}$ ▷ 计算学生网络 Jacobian 矩阵
$\boldsymbol{J^i_{teacher}\leftarrow J(z^i_{teacher}, x, M_j)}$ ▷ 计算教师网络 Jacobian 矩阵
$\boldsymbol{L^i_j\leftarrow\left(J^i_{student}-J^i_{teacher}\right)^2}$ ▷ 计算教师网络与学生网络 Jacobian 矩阵损失
**end**
$\boldsymbol{Loss\leftarrow CE(x,y)+\sum_{i=0}^{n}\sum_{j=0}^{n_{proj}}L^i_j}$ ▷ 更新总损失函数
更新 $\boldsymbol{f_{student}}$ 参数
**end**
**return** $\boldsymbol{f_{student}}$
**end**

### 3.4 小结

我们的方法是基于特征图的约束与基于 Jacobian 矩阵约束的结合。特征图的约束可以很好地使学生网络保留教师网络的特征, 而 Jacobian 的加入在梯度上进一步约束网络的求解, 最终的损失函数为

$$L(x,y)=CE(x,y)+\alpha L_{fm}+\beta L_{jacobian} \quad (21)$$

$\alpha$ 与 $\beta$ 为正则项损失函数的权重。

## 4 实验结果及其分析

我们在 Cifar10、Cifar100、ImageNet 上进行了

大量的实验, 将从实验设置、基于特征图与 Jacobian 矩阵约束的作用、约束特征层深度影响、Cosine 距离的作用、损失函数权重的影响、$n_{proj}$选取、不同数据集上算法效果等方面论述我们提出的方法的有效性。所有实验均在白盒攻击场景下进行。

## 4.1 实验设置

本部分工作主要是为了验证我们的算法在模型鲁棒性迁移上的效果, 我们的教师网络与学生网络均保持一致, 使用的是 ResNet50[22], 这里的网络是针对 32*32 大小的图像重新设计的网络, 在 Cifar10、Cifar100、32*32 大小的 ImageNet 上可以获得非常不错的效果。

在本部分中, 为了保证算法的泛化性尽可能强, 我们在评估评估算法的时候, 使用了多种攻击算法, 包含 FGSM, $L_2$ PGD 攻击算法, $L_\infty$ PGD 攻击算法, 这些算法均为无目标攻击。对于$L_\infty$ 攻击算法, 我们攻击的最大扰动分别选取了 $\frac{4}{255}$ 与 $\frac{8}{255}$, 对于 $L_2$ 攻击算法, 我们攻击的最大扰动分别选取了 0.25 与 0.5, 对于迭代式的攻击, 假设最大扰动为 $\epsilon$, 攻击的步长为 $\frac{\epsilon}{5}$, 如无特殊说明, 本部分的实验 PGD 攻击算法的迭代次数均为 7。

对于鲁棒性教师网络, 我们分别在 ImageNet、Cifar100 这两个数据集上, 通过对抗训练的方法得到了两个鲁棒的模型, 对抗训练中使用的攻击方法是 $L_2$ PGD 攻击算法, 攻击最大扰动为 0.5, 训练使用的是带有 momentum 的 SGD[23]算法, momentum 为 0.9, weight decay 为 5e-4, 在 ImageNet 数据集上, 我们会将图片变换成 32*32 的大小。

对于学生网络, 学生网络在训练时的输入均为干净的样本, 教师网络的参数固定, 学生网络训练的超参数与教师网络保持一致, 为了加快训练速度, 学生网络均使用了对抗训练后的教师网络作为预训练模型, $\alpha$ 选取为 50, $\beta$ 选取为 5e-3。

本文所有实验均在 Pytorch[24]框架下实现, 本部分会涉及算法效率的比对, 所有的实验均在 Tesla v100-sxm2 显卡下运行。

## 4.2 算法效果

这部分在 Cifar10 数据上对比算法效果, 通过准确率来评价模型鲁棒性的强弱, 如果该模型能够将越多的对抗样本分类正确, 那么则说明该模型的鲁棒性越强, 防御对抗样本的能力越强。

我们首先在 Cifar10 上以正常训练以及对抗训练两种方式, 训练了两个模型作为我们的基线, 为了对比公平, 这两个模型同样也使用了预训练模型。我们在同样的设置下, 复现了 Shafahi[7]的工作, 通过与 Shafahi[7]工作的对比, 来体现我们算法优越性。我们同样比对了分为去除掉基于特征的约束项与去除掉基于 Jacobian 矩阵的约束项的效果, 在本部分中 $n_{proj}$ 为 1。在 PGD 攻击上的效果如下表 1 所示。

**表 1 PGD 攻击算法在 Cifar10 上的分类准确率(%)**

**Table 1 Accuracy(%) of PGD attack on Cifar10**

| 模型 | Clean | $L_2$ PGD | | $L_\infty$ PGD | | Time |
|---|---|---|---|---|---|---|
| | | 0.25 | 0.5 | 4/255 | 8/255 | |
| 正常训练 | 97.03 | 27.86 | 3.54 | 1.82 | 0.01 | ~25 min |
| 对抗训练 | 92.19 | **83.80** | **72.12** | **68.05** | **37.71** | ~6 h |
| Shafahi'swork[7] | 96.14 | 73.61 | 39.44 | 34.01 | 4.89 | ~35 min |
| Robust-KDw/o $L_{fm}$ | 94.27 | 76.43 | 46.92 | 40.44 | 6.23 | ~3 h |
| Robust-KDw/o $L_j$ | 95.45 | 82.42 | 56.19 | 50.93 | 11.95 | ~35 min |
| Robust-KD | 94.83 | **83.70** | **62.02** | **55.89** | **16.20** | ~3.5 h |

通过对比, 可以发现我们的算法在对于鲁棒性蒸馏的效果非常好, 在受限的训练时间下, 可以获得接近对抗训练的防御效果, 此外我们的防御方法在干净样本上依然保留了很高的准确率。相较于 Shafahi 等人[7]的工作, 我们的鲁棒性迁移的效果更强, 可以将更多的对抗样本识别正确。因为计算 Jacobian 矩阵要进行多次反向传播, 所以虽然带入 Jacobian 约束的防御方法在防御效果上有一定提升, 但是需要耗费非常多的计算资源。

同样, 我们还在 FGSM 攻击算法上测试了我们的防御模型, 如下表 2 所示。

在 FGSM 下, 我们的算法同样有非常不错的防御效果, 在扰动为 $\frac{4}{255}$ 的 FGSM 攻击算法下, 可以获得接近对抗训练的效果。此外, 在更加强大的 PGD 攻击下, 我们测试了模型的效果。一般来说, PGD 攻击算法的迭代次数越大, 相应的攻击越强, 此处我们分别测试了迭代次数为 0、7、25、100、250 与 1000 的白盒攻击下, 模型的准确率。下图 4 展示了模型防

表 2 FGSM 攻击算法在 Cifar10 上的分类准确率(%)

Table 2 Accuracy(%) of FGSM attack on Cifar10

| 模型 | Clean | FGSM | |
|---|---|---|---|
| | | 4/255 | 8/255 |
| 正常训练 | 97.03 | 23.17 | 16.12 |
| 对抗训练 | 92.19 | 71.28 | 51.36 |
| Shafahi's work[7] | 96.14 | 56.16 | **34.29** |
| Robust-KD | 94.83 | **62.42** | 31.27 |

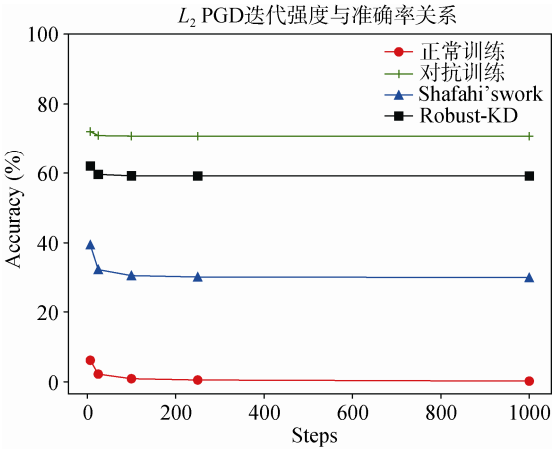


图 4 $L_2$ PGD 迭代强度与准确率(%)的关系

Figure 4 The relationship between steps of $L_2$ PGD attack and accuracy (%)

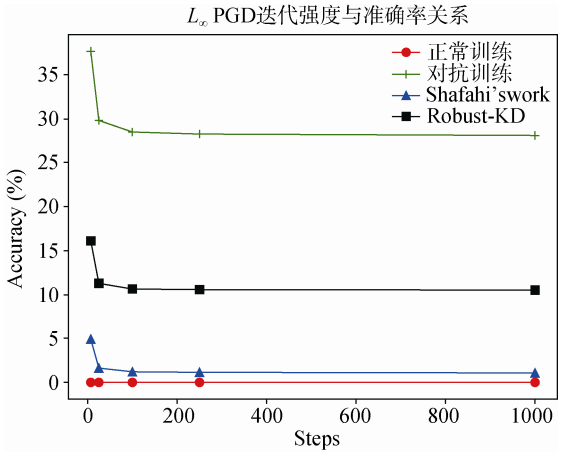


图 5 $L_\infty$ PGD 迭代强度与准确率(%)的关系

Figure 5 The relationship between steps of $L_\infty$ PGD attack and accuracy (%)

御 0.5 扰动下不同迭代次数的 $L_2$ PGD 攻击的效果，下图 5 展示了模型防御 $\frac{8}{255}$ 扰动下不同迭代次数的 $L_\infty$ PGD 攻击的效果。

甚至在 1000-step 的 PGD 攻击下，我们的模型依然保留了可观的准确率。

### 4.3 消融实验

在本部分中，我们进行了一系列的消融实验，对比分析各个模块的作用。

#### 4.3.1 不同特征层约束

ResNet50 共有 4 个 res-block，4.2 中的实验对所有的 res-block 输出的特征图上都添加了约束，这样对神经网络的浅层特征以及深层特征都有非常好的约束，可以大大提升算法效果。

本部分想要分析约束在 res-block 上的效果，实验设置与 4.2 保持一致，$n_{proj}$ 为 1，下表 3 中给出了实验结果。

可以发现，随着添加约束的特征层越来越深，鲁棒性迁移的效果越好，仅在最后一层添加正则的防御能力与在所有层添加正则的模型效果相当。

#### 4.3.2 损失函数权重

本部分对损失函数权重的选取进行对比实验，考虑到我们的方法包含两个正则项，这里我们分两

表 3 约束层对于 PGD 攻击算法在 Cifar10 准确率(%)的影响

Table 3 The influence of the constraint layer on the accuracy (%) of the PGD attack on Cifar10

| 约束层 | Clean | $L_2$ PGD | | $L_\infty$ PGD | |
|---|---|---|---|---|---|
| | | 0.25 | 0.5 | 4/255 | 8/255 |
| First block | 96.21 | 50.24 | 12.70 | 9.51 | 0.34 |
| Second block | 96.12 | 72.62 | 37.67 | 33.21 | 4.28 |
| Third block | 95.56 | 80.64 | 55.19 | 49.84 | 12.66 |
| Fourth block | 94.38 | 82.62 | **62.89** | **56.32** | **18.02** |
| All blocks | **94.83** | **83.70** | 62.02 | 55.89 | 16.20 |

表 4 基于不同权重特征图约束的鲁棒性迁移在 Cifar10 上的准确率(%)

Table 4 Accuracy(%) of robustness transfer based on feature map constraints under different weights on Cifar10

| 权重 $\alpha$ | Clean | $L_2$ PGD | | $L_\infty$ PGD | |
|---|---|---|---|---|---|
| | | 0.25 | 0.5 | 4/255 | 8/255 |
| 5e-1 | 96.86 | 43.21 | 8.56 | 6.18 | 0.21 |
| 5 | 96.34 | 70.37 | 32.59 | 29.42 | 3.32 |
| 5e-1 | **95.45** | 82.42 | 56.19 | 50.93 | 11.95 |
| 5e-2 | 94.90 | **82.52** | **59.84** | **53.04** | **15.24** |
| 5e-2 | 91.86 | 76.26 | 52.18 | 44.69 | 10.41 |

部分进行实验。一部分是基于特征图的正则项的权重选择; 另一部分是基于 Jacobian 矩阵的正则项的权重选择。基于特征图的鲁棒性迁移的消融实验结果见下表 4。

当 $\alpha$ 为 5e-1 或者 5e-2 时, 在没有损失过多干净样本准确率的前提下, 模型防御能力最强。基于 Jacobian 矩阵的鲁棒性迁移的消融实验结果见下表 5。

表 5 基于不同权重 Jacobian 约束的鲁棒性迁移在 Cifar10 上的准确率(%)

Table 5 Accuracy(%) of robustness transfer based on Jacobian constraints under different weights on Cifar10

| 权重 $\beta$ | Clean | $L_2$ PGD | | $L_\infty$ PGD | |
|---|---|---|---|---|---|
| | | 0.25 | 0.5 | 4/255 | 8/255 |
| 5 | 96.85 | 43.08 | 7.58 | 5.51 | 0.10 |
| 5e-1 | 95.62 | 65.00 | 23.97 | 18.75 | 0.78 |
| 5e-2 | 95.06 | 74.95 | 40.99 | 35.42 | 4.24 |
| 5e-3 | **94.27** | 76.43 | 46.92 | 40.44 | 6.23 |
| 5e-4 | 93.59 | **80.05** | **57.85** | **51.48** | **14.42** |
| 5e-5 | 80.57 | 62.56 | 40.15 | 34.47 | 6.53 |

当 $\beta$ 为 5e-3 或者 5e-4 时, 在没有损失过多干净样本准确率的前提下, 模型防御能力最强。

### 4.3.3 Cosine 距离的效果

在基于特征图的鲁棒性迁移部分中, 我们使用了 Cosine 矩阵作为我们的约束方式而非 $L_2$ 距离。在本部分实验中, 我对特征图之间的 $L_2$ 损失函数进行了一些实验, 本部分并没有引入 Jacobian 矩阵约束。我们不断调整 $L_2$ 损失函数的权重, 实验效果如下表 6 所示。

从上表中, 可以看出相较于 $L_2$ 距离, Cosine 距离在约束特征图上效果略好。

### 4.3.4 $n_{proj}$ 的选取

在本部分, 我们验证了在 Jacobian 约束中 $n_{proj}$ 的作用, 考虑到效率问题, 在本文的大部分实验上, 我们均使用 $n_{proj}=1$ 的设定。在本部分中, 我们验证了 $n_{proj}$ 增大的效果, 在实验中, 考虑到复杂度, 本部分实验的约束仅添加在最后一个 res-block, 其仅使用了 Jacobian 矩阵作为鲁棒性迁移的媒介, 实验结果见下表 7。

表 6 Cosine 距离与 $L_2$ 距离在 Cifar10 上的准确率(%)的对比

Table 6 Accuracy(%) on Cifar10 under Cosine distance and $L_2$ distance

| 约束方式 | Clean | $L_2$ PGD | | $L_\infty$ PGD | |
|---|---|---|---|---|---|
| | | 0.25 | 0.5 | 4/255 | 8/255 |
| 5e-1 $L_2$ | 96.69 | 32.90 | 4.88 | 3.07 | 0.04 |
| 5e-0 $L_2$ | 96.30 | 72.83 | 37.62 | 33.01 | 4.73 |
| 5e-1 $L_2$ | 94.77 | 80.22 | 57.54 | 51.85 | 15.16 |
| 5e-2 $L_2$ | 93.11 | 79.72 | 57.14 | 51.29 | 13.65 |
| 5e-3 $L_2$ | 90.92 | 73.24 | 48.52 | 40.50 | 8.54 |
| Cosine | **94.90** | **82.52** | **59.84** | **53.04** | 15.24 |

表 7 不同 $n_{proj}$ 下 PGD 攻击算法在 Cifar10 上的准确率(%)

Table 7 Accuracy (%) of the PGD attack on Cifar10 under different $n_{proj}$

| $n_{proj}$ | Clean | $L_2$ PGD | | $L_\infty$ PGD | |
|---|---|---|---|---|---|
| | | 0.25 | 0.50 | 4/255 | 8/255 |
| 1 | 92.76 | 76.59 | 50.87 | 44.07 | 9.23 |
| 5 | 92.82 | 75.60 | 49.73 | 43.84 | 9.69 |
| 10 | **92.97** | 76.81 | **51.86** | **44.95** | **10.17** |

随着 $n_{proj}$ 的增加, 算法效果有一定程度上的提升, 但是带来的提升非常有限, 而时间复杂度增长非常多, 所以在本文的实验中, 我们使用 $n_{proj}=1$ 的设置。

#### 4.3.5 算法泛化性验证

本部分之前的教师模型是在 ImageNet 上训练的, 目标数据集是 Cifar10, 无法证明当前我们算法泛化能力。所以在本部分中, 我们分别测试了 ImageNet 迁移 SVHN, ImageNet 迁移 Cifar100, Cifar100 迁移 Cifar10 等情况, 证明我们的方法在不同源数据集与目标数据集上, 均有客观的效果。算法效果见下表 8。

可以看出, ImageNet 迁移 Cifar10 的效果要好于 Cifar100 迁移 Cifar10, 教师网络的效果好坏, 一定程度上也会影响学生网络的好坏。在 Cifar100 数据上的效果也证明了我们鲁棒性迁移算法的泛化能力, 在不同数据集上, 可以获得不错的效果。但是在 SVHN 的数据集上, 虽然 ImageNet 迁移 SVHN 依然有着不错的效果, 但是与 ImageNet 迁移 Cifar10、Cifar100 相比, 效果要差了不少。因为 Cifar100 数据集与 SVHN 存在非常低的相似性, 所以迁移地效果非常不理想, 但这同样也验证了在 ImageNet 中提取的鲁棒性特征的泛化性是不错的。

#### 4.3.6 同源数据集之间的迁移效果

对于上述实验, 我们验证的是不同数据之间的迁移效果, 在迁移学习中, 源数据集与目标数据集之间的相关性也是衡量算法指标的重要考量因素。考虑到上一节中的现象, 在本节我们将验证同源数据集之间鲁棒性迁移的效果。

在本节, 我们将数据集平均切分为两部分, 例如 SVHN 的训练集切分为 SVHN+、SVHN-, 在 SVHN+与 SVHN-之间迁移模型的鲁棒性。同源数据集之间的鲁棒性迁移效果如下表所示。

表 9 的结果显示, 同源数据集之间的鲁棒性迁移效果非常接近对抗训练, 尽管 SVHN+或者 Cifar10+的数据量远不及 ImageNet, 但是与目标数据集之间的相似度更高, 防御效果也更强。因此, 在鲁棒性迁移中, 源数据集与目标数据集之间的相似度也是需要考虑一个方面, 在后续的研究工作中, 我们也将深入研究这个现象。

表 8 不同源与目标数据集下 PGD 攻击算法的准确率(%)

Table 8 Accuracy (%) of the PGD attack under different source and target datasets

| 模型 | 迁移方式 | Clean | $L_2$ PGD | |
|---|---|---|---|---|
| | | | 0.25 | 0.50 |
| 正常训练 | SVHN | 96.94 | 54.69 | 26.30 |
| 对抗训练 | SVHN | 95.24 | 86.73 | 71.68 |
| Robust-KD | Cifar100→SVHN | 95.13 | 52.73 | 20.25 |
| Robust-KD | ImageNet→SVHN | **96.45** | **69.70** | **40.48** |
| 正常训练 | Cifar10 | 97.03 | 27.86 | 3.54 |
| 对抗训练 | Cifar10 | 92.19 | 83.80 | 72.12 |
| Robust-KD | Cifar100→Cifar10 | 89.77 | 77.45 | 57.86 |
| Robust-KD | ImageNet→Cifar10 | **94.83** | **83.70** | **62.02** |
| 正常训练 | Cifar100 | 83.31 | 12.35 | 3.20 |
| 对抗训练 | Cifar100 | 73.44 | 60.67 | 47.67 |
| Robust-KD | ImageNet→Cifar100 | 77.52 | 58.97 | 37.83 |

**表 9　同源与目标数据集下 PGD 攻击算法的准确率(%)**

**Table 9　Accuracy (%) of the PGD attack under same source and target datasets**

| 模型 | 迁移方式 | Clean | $L_2$ PGD | |
|---|---|---|---|---|
| | | | 0.25 | 0.50 |
| 正常训练 | SVHN- | 96.27 | 51.71 | 22.03 |
| 对抗训练 | SVHN- | 94.59 | 85.57 | 69.88 |
| Robust-KD | ImageNet→SVHN- | 95.67 | 66.86 | 35.39 |
| Robust-KD | SVHN+→SVHN- | **96.28** | **83.56** | **60.37** |
| 正常训练 | Cifar10- | 95.93 | 30.60 | 4.59 |
| 对抗训练 | Cifar10- | 91.75 | 83.27 | 70.88 |
| Robust-KD | ImageNet→Cifar10- | 94.75 | 80.65 | 56.29 |
| Robust-KD | Cifar10+→Cifar10- | **93.85** | **84.33** | **68.91** |

## 5　结论

在本文中我们提出了一种鲁棒性迁移的方法, 用来解决对抗训练高时间复杂度的问题。在我们的工作中, 可以从一个鲁棒的教师网络中, 将鲁棒性迁移到一个网络中, 通过这种迁移方法, 可以以比较低的代价, 获得一个防御能力相对较强的网络。

我们的工作从基于特征图的正则与基于 Jacobian 矩阵的正则出发, 通过这两种约束, 使神经网络在新的任务上, 依然保留了一部分教师网络中的鲁棒性特征。我们进行了相关实验, 证明了我们提升的算法的有效性。

我们的算法还有很多可以改进的方向, 一是如何进一步优化 Jacobian 矩阵的约束, 进一步降低算法的耗时, 二是可以更加深入地分析神经网络的性质, 针对鲁棒性模型的特性, 设计更加有效的迁移方法。此外, 在原数据与目标数据集之间的相关性还有更多可以挖掘的地方。在后续的工作中, 我们将在这些方面优化改进。

## 参考文献


[1] LeCun Y, Bengio Y, Hinton G. Deep Learning[J]. *Nature*, 2015, 521(7553): 436-444.

[2] Lin C H, Chang C C, Chen Y S, et al. COCO-GAN: Generation by Parts via Conditional Coordinating[C]. *2019 IEEE/CVF International Conference on Computer Vision*, 2019: 4511-4520.

[3] Touvron H, Vedaldi A, Douze M, et al. Fixing the train-test resolution discrepancy[C]. *Advances in Neural Information Processing Systems*, 2019: 8252-8262.

[4] Szegedy C, Zaremba W, Sutskever I, et al. Intriguing properties of neural networks[J]. arXiv preprint arXiv:1312.6199, 2013.

[5] Eykholt K, Evtimov I, Fernandes E, et al. Robust Physical-World Attacks on Deep Learning Visual Classification[C]. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018: 1625-1634.

[6] Madry A, Makelov A, Schmidt L, et al. Towards Deep Learning Models Resistant to Adversarial Attacks[EB/OL]. 2017.

[7] Shafahi A, Saadatpanah P, Zhu C, et al. Adversarially Robust Transfer Learning[EB/OL]. 2019.

[8] Hinton G, Vinyals O, Dean J. Distilling the knowledge in a neural network[J]. arXiv preprint arXiv:1503.02531, 2015.

[9] Zagoruyko S, Komodakis N. Paying more Attention to Attention: Improving the Performance of Convolutional Neural Networks via Attention Transfer[EB/OL]. 2016.

[10] Etmann C, Lunz S, Maass P, et al. On the Connection between Adversarial Robustness and Saliency Map Interpretability[EB/OL]. 2019.

[11] Pan S J, Yang Q. A Survey on Transfer Learning[J]. *IEEE Transactions on Knowledge and Data Engineering*, 2010, 22(10): 1345-1359.

[12] Deng J, Dong W, Socher R, et al. ImageNet: A Large-Scale Hierarchical Image Database[C]. *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009: 248-255.

[13] Goodfellow I J, Shlens J, Szegedy C. Explaining and harnessing adversarial examples[J]. arXiv preprint arXiv:1412.6572, 2014.

[14] Zhang H C, Wang J Y. Defense Against Adversarial Attacks Using Feature Scattering-Based Adversarial Training [C]. Advances in Neural Information Processing Systems, 2019: 1831-1841.

[15] Zhang H Y, Yu Y D, Jiao J T, et al. Theoretically Principled Trade-off between Robustness and Accuracy[EB/OL]. 2019.

[16] Ross A S, Doshi-Velez F. Improving the Adversarial Robustness and Interpretability of Deep Neural Networks by Regularizing Their Input Gradients[EB/OL]. 2017.

[17] Hoffman J, Roberts D A, Yaida S. Robust Learning with Jacobian Regularization[EB/OL]. 2019.

[18] Le Y, Yang X. Tiny imagenet visual recognition challenge[J]. CS 231N, 2015, 7.

[19] Li Z, Hoiem D. Learning without forgetting[J]. IEEE transactions on pattern analysis and machine intelligence, 2017, 40(12): 2935-2947.

[20] Goldberger, Gordon, Greenspan. An Efficient Image Similarity Measure Based on Approximations of KL-Divergence between Two Gaussian Mixtures[C]. *Ninth IEEE International Conference on Computer Vision*, 2003: 487-493.

[21] Qian G, Sural S, Gu Y, et al. Similarity between Euclidean and cosine angle distance for nearest neighbor queries[C]. *The 2004 ACM symposium on Applied computing*, 2004: 1232-1237.

[22] He K M, Zhang X Y, Ren S Q, et al. Deep Residual Learning for Image Recognition[C]. *2016 IEEE Conference on Computer Vision and Pattern Recognition*, 2016: 770-778.

[23] Bottou L. Large-Scale Machine Learning with Stochastic Gradient Descent[C]. *Proceedings of COMPSTAT'2010*, 2010: 177-186.

[24] Paszke A, Gross S, Chintala S, et al. Automatic differentiation in pytorch[C]. *Advances in Neural Information Processing Systems Workshop,* 2017.

[25] Dong Y P, Liao F Z, Pang T, et al. Boosting Adversarial Attacks with Momentum[C]. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018: 9185-9193.

[26] Carlini N, Wagner D. Towards Evaluating the Robustness of Neural Networks [C]. *2017 IEEE symposium on security and privacy*, 2017: 39-57.

[27] Qin C L, Martens J, Gowal S, et al. Adversarial Robustness through Local Linearization[EB/OL]. 2019.

[28] Drucker H, Le Cun Y. Double Backpropagation Increasing Generalization Performance[C]. *IJCNN-91-Seattle International Joint Conference on Neural Networks*, 1991: 145-150.

[29] Chan A, Tay Y, Ong Y S, et al. Jacobian Adversarially Regularized Networks for Robustness[EB/OL]. 2019.

[30] Hein M, Andriushchenko M. Formal guarantees on the robustness of a classifier against adversarial manipulation[C]. *Advances in Neural Information Processing Systems*, 2017: 2266-2276.

[31] Wong E, Schmidt F, Metzen J H, et al. Scaling Provable Adversarial Defenses[EB/OL]. 2018.

[32] Ma X J, Li B, Wang Y S, et al. Characterizing Adversarial Subspaces Using Local Intrinsic Dimensionality[EB/OL]. 2018.

[33] Krishna K, Murty M N. Genetic K-means algorithm[J]. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 1999, 29(3): 433-439.

[34] Meng D, Chen H. Magnet: a two-pronged defense against adversarial examples[C]. The 2017 ACM SIGSAC conference on computer and communications security, 2017: 135-147.

[35] Xu W L, Evans D, Qi Y J. Feature Squeezing: Detecting Adversarial Examples in Deep Neural Networks[EB/OL]. 2017.

[36] 王科迪, 易平. 人工智能对抗环境下的模型鲁棒性研究综述[J]. *信息安全学报*, 2020, 5(3): 13-22.

**张维** 于 2018 年在上海交通大学信息安全 专业获得学士学位。现在上海交通大学电子与通信工程专业攻读硕士研究生学位。研究领域为对抗样本与人工智能安全。研究兴趣包括:对抗样本防御。Email: mercurialzhang@163.com

**易平** 于 2005 年复旦大学计算机应用专业获得博士学位。上海交通大学网络空间安全学院副教授。研究领域为网络对抗。研究兴趣包括: 人工智能安全。Email: yiping@sjtu.edu.cn