# 防御模型实验报告

张哲昊　　王子龙
519030910383　　51903091038

日期：2021 年 6 月 13 日

**摘　　要**

本次防御任务，我们小组从防御模型的训练方法着手，寻找文献资料，学习了对抗攻击，防御蒸馏，温度计编码等一系列防御方法。接着着重于对抗攻击对 baseline 进行进一步优化，包括损失函数的调整，训练过程的加速，正则化的应用，其他训练技巧的尝试。接下来尝试应用额外数据集进行预训练。使用 Resnet18 和 WideResnet28 模型用任务一中的进攻方法应用于对抗训练，通过对比实验，收获颇丰。

## 1　防御任务整体设定

### 1.1　对抗训练

对抗训练（adversarial training）是增强神经网络鲁棒性的重要方式。在对抗训练的过程中，样本会被混合一些微小的扰动（改变很小，但是很可能造成误分类），然后使神经网络适应这种改变，从而对对抗样本具有鲁棒性。但是对抗训练的方法仍然有其局限性，通常当模型对于干扰之后的图像有了更加鲁棒的结果，但是对于普通的图像的分类准确率却并不令人满意。所以在对抗训练过程中需要对准确度和鲁棒性进行 trade-off。

对抗训练的原理由以下公式呈现：

$$\min_{\theta} \mathbb{E}_{(Z,y)\sim\mathcal{D}}\left[\max_{||\delta||\leq\epsilon} L(f_\theta(X+\delta),y)\right]$$

内层的最大化是为了在扰动在一定约束的情况下，找到最能够“迷惑”该模型的扰动。外层的最小化问题是为了使得当扰动固定的情况下，我们训练神经网络模型使得在训练数据上的损失最小，也就是说，使模型具有一定的鲁棒性能够适应这种扰动。

### 1.2　其他训练方式

对抗训练并不是防御对抗攻击唯一的方法，还有诸如 Defensive Distillation、Thermometer Encoding、Pixel Defend、对输入进行随机化处理等其他方法。

Defensive Distillation(防御蒸馏)：在这个防御方法中，我们会选择两个相同的模型作为教师模型和学生模型。首先，用硬标签训练教师模型，此时温度设置为 T。然后用教师模型输出类别概率

**Algorithm 1:** Simple black-box attack

**Result:** Robust model

**Initialization:** $x = x_{input} \quad model = model_{input} \quad y = label$

$loss = cross\_entropy(model(x), y)$;

$x_{adv} = attack(model, x, y)$

$ouput_{adv} = model(x_{adv})$

Loss=loss($ouput_{adv}, y$)

loss.backward

return model

f(x)，将这个概率作为软标签（soft label）来训练学生模型，温度同样为 T。最终输出预测时，使用学生模型输出预测，此时温度置为 1, 更改温度是改变最后的 softmax 输出层，增大 T 会使最后的输出更加 soft。利用软标签训练模型可以更好的去拟合不同类别之间的相似性，从而抵御攻击。

Thermometer Encoding(温度计编码)：类似于温度计的表示，将一个元素 x 张成一个长为 l 的向量，当 x<i/l 时 $x_i = 1$，其他情况 $x_i = 0$，这实质上是一种梯度掩蔽策略，在这一层无法求出梯度。

pixel defend：将攻击图像进行滤波处理，具体是将其转变为在 training data 中出现过的分布，再将滤波后的图像通入分类器模型进行分类。

# 2 防御任务 baseline 相关改进

## 2.1 Loss function 的调整与正则化

Baseline 中的交叉熵损失函数作为分类任务最常见且最有效的损失函数并不一定是对抗训练中最适合的损失函数，因为其只是以识别准确率为标准计算损失而并没有考虑模型在噪声情况下的鲁棒性。通过参考此前攻击任务的防御模型，我们将损失函数换成 TRADES，泛化意义下的公式如下：

$$\mathcal{L} = \alpha_1 \cdot L_1(x, y) + \alpha_2 \cdot L_2(x_{adv}, y)$$

其中前一项代表了对于模型的识别准确度的刻画，而后一项的代表对模型的抗干扰性的刻画，$\alpha_1$ 与 $\alpha_2$ 代表了对两项的相对的 “偏好”。而后一项也相当于对损失函数进行了一定的正则化，可以在一定程度上可以缓解过拟合。本项目当中损失函数的设计如下：

$$\mathcal{L} = \alpha_1 \cdot cross_(x, y) + \alpha_2 D_{KL}(Softmax(x), Softmax(x_{adv}))$$

## 2.2 训练中加入随机化

## 2.3 加速对抗训练

为了加速对抗样本的生成过程以便于训练，我们应用速度非常快的 FGSM 攻击（具体算法在攻击报告里可见），假设损失函数在样本点处为局部线性，快速生成 $l_\infty$ 范数限制下的对抗样

本：

$$x_{adv} = x + \epsilon \cdot sign(\nabla_x L(\theta, x, y))$$

该算法能够极大地加快训练速度，相比于 baseline 可以提高约 5 倍的速度，但由于攻击样本的性能限制，最终模型的鲁棒性不太理想（但准确率较高）。当重新 TRADES loss 的参数 $\alpha_2$ 后，模型的鲁棒性会得到一定的提升。

### 2.4 正则化

## 3 训练技巧

参考 Bag of Tricks for Adversarial Training (ICML2021) 这篇论文中提到的训练技巧，本次项目尝试进行部分实现。

i) Early stopping w.r.t. training epochs　该训练技巧能够防止因训练 epoch 过多而产生过拟合现象。

ii) Warmup w.r.t. learning rate , 模型的权重 (weights) 是随机初始化的，此时若选择一个较大的学习率, 可能带来模型的不稳定 (振荡)，选择 Warmup 预热学习率的方式，可以使得开始训练的几个 epoches 或者一些 steps 内学习率较小, 在预热的小学习率下，模型可以慢慢趋于稳定, 等模型相对稳定后再选择预先设置的学习率进行训练, 使得模型收敛速度变得更快，模型效果更佳。

iii) Label smoothing 标签平滑后的分布就相当于往真实分布中加入了噪声，避免模型对于正确标签过于自信，使得预测正负样本的输出值差别不那么大，从而避免过拟合，提高模型的泛化能力。

iv) Weight decay 相当于二范数正则化，也可以解决过拟合问题。

v) Activation function 可以把模型里的激活函数换成光滑版本的 Relu 如 Softplus（$f(x) = \ln(1 + e^x)$）和 GELU（$GELU(x) \approx 0.5x\left\{1 + \tanh\left[\sqrt{2/\pi}(x + 0.044715x^3)\right]\right\}$），效果会更好。

## 4 额外数据集的加入

本次实验的数据集为 cifar10，同时我们尝试增加其他数据集进行训练，经过资料查找，尝试了使用 fashion minist 数据集。该数据集同样是 10 类的图像，但是为灰度图像，故不符合网络结构要求。同时考虑使用 STL10 数据集，但是该数据集大部分为无标签的数据，不太适合本次监督学习的任务。我们还尝试使用 Imagenet，但是其数据量过大，且包含 1000 个类的图像。而 cifar100 包含 100 个类的数据，同样不太适合本次实验。

# 5 实验结果汇总

| 模型 | 攻击方法 | 损失函数 | 训练时长 | 准确率 | 攻击后准确率 |
|---|---|---|---|---|---|
| ResNet18 | PGD | 交叉熵 | 15 小时 | 0.8357 | 0.5138 |
| ResNet18 | FGSM | 交叉熵 | 6 小时 | 0.89570 | 0.16600 |
| ResNet18 | FGSM | TRADES | 10 小时 | 0.90160 | 0.27280 |
| ResNet18 | my attack | 交叉熵 | 40 小时 | 0.83360 | 0.53070 |
| WideResNet28 | FGSM | 交叉熵 | 18 小时 | 0.90430 | 0.11240 |
| ResNet18 | my attack | TRADES | 50 小时 | 0.80940 | 0.55150 |

运行实验的硬件条件：CPU:AMD R9 5900HX　GPU:NVIDIA RTX3080 laptop（单卡）
由于只有一张卡，wideresnet28 模型运行过慢，但是 resnet18 上的成绩已经超过 baseline 10% 左右，故可以认为模型效果不错。My attack 的平均正向传播次数：782　反向传播次数：391

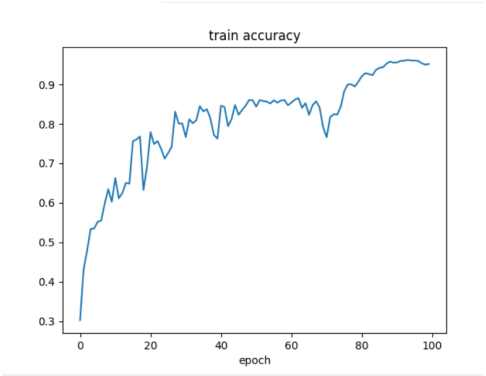


**图 1:** FGSM attack with cross entropy Resnet18

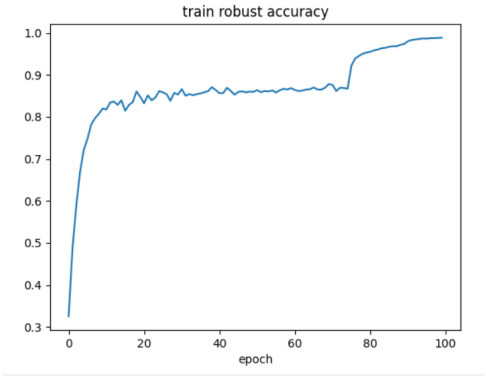


**图 2:** FGSM attack with cross entropy Resnet18

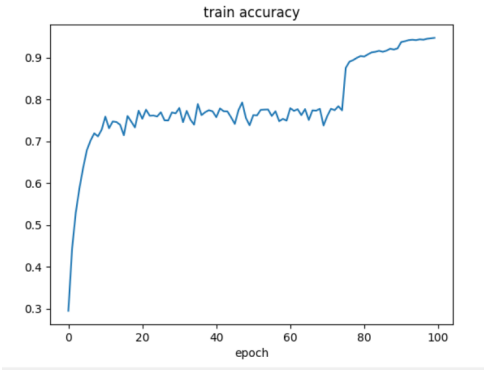


**图 3:** FGSM attack with TRADES Resnet18

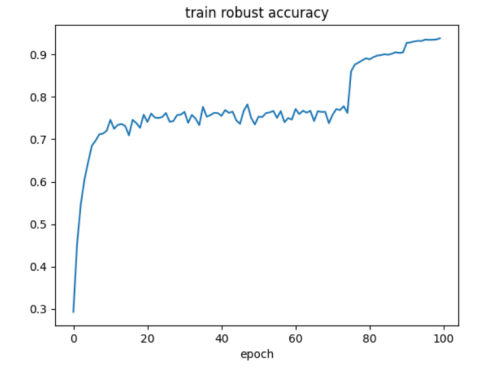


**图 4:** FGSM attack with TRADES Resnet18

# 6 致谢

感谢时若曦小组对我们的工作进行讨论与交流，以及江学姐的答疑。

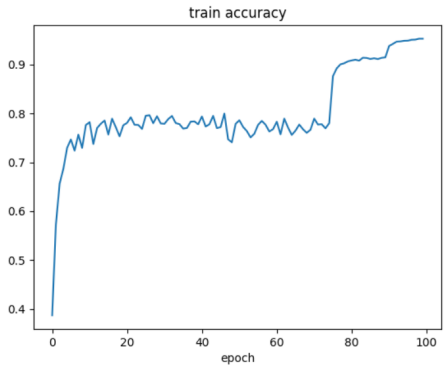


**图 5:** FGSM attack with TRADES wideResnet28

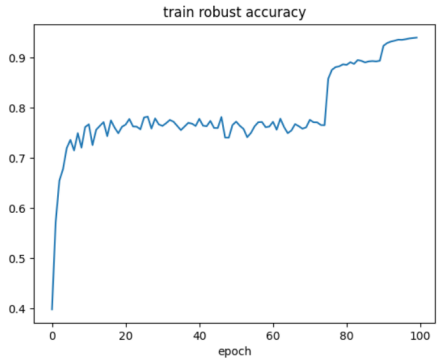


**图 6:** FGSM attack with TRADES wideResnet28