哈尔滨工业大学(深圳)

《网络与系统安全》 实验报告

实验七

对抗样本攻击 实验

学 院: 计算机科学与技术

姓 名: 梁鑫嵘

学 号: 200110619

专 业: 计算机科学与技

日 期: 2023 年 4 月

一、本次实验要求

1. 完成 **4.3 FGSM 攻击函数**的代码补充，截图说明。

```python
# FGSM attack code
def fgsm_attack(image, epsilon, data_grad):
    # Collect the element-wise sign of the data gradient
    sign_data_grad = data_grad.sign()

    # Create the perturbed image by adjusting each pixel of the input image
    perturbed_image = image + epsilon * sign_data_grad

    # Adding clipping to maintain [0,1] range
    perturbed_image = torch.clamp(perturbed_image, 0, 1)

    # Return the perturbed image
    return perturbed_image
```

[5] ✓ 0.0s

2. 分析 **4.4 测试攻击效果函数** 的代码部分，说明每段代码的作用。

这段代码实现了一个用于对模型进行攻击的测试函数。

1. 函数名为 test，接受以下参数：
   - model：要测试的模型
   - device：设备（如 CPU 或 GPU）用于计算

- test_loader：测试数据集的数据加载器
- epsilon：FGSM 攻击中的扰动大小

2. 在函数内部初始化了计数器 correct 和存储对抗样本的列表 adv_examples。
3. 使用 for 循环遍历测试集中的所有样本。
4. 将数据和标签发送到指定设备。
5. 设置 requires_grad 属性为 True，以便在攻击中计算梯度。
6. 将数据通过模型进行前向传播，得到预测结果 output 和初始预测值 init_pred。
7. 如果初始预测与真实标签不一致，跳过此样本。
8. 计算损失 loss，使用负对数似然损失函数（negative log-likelihood loss）。
9. 清零模型的所有梯度。
10. 在反向传播过程中计算模型的梯度。
11. 收集梯度数据 data_grad。

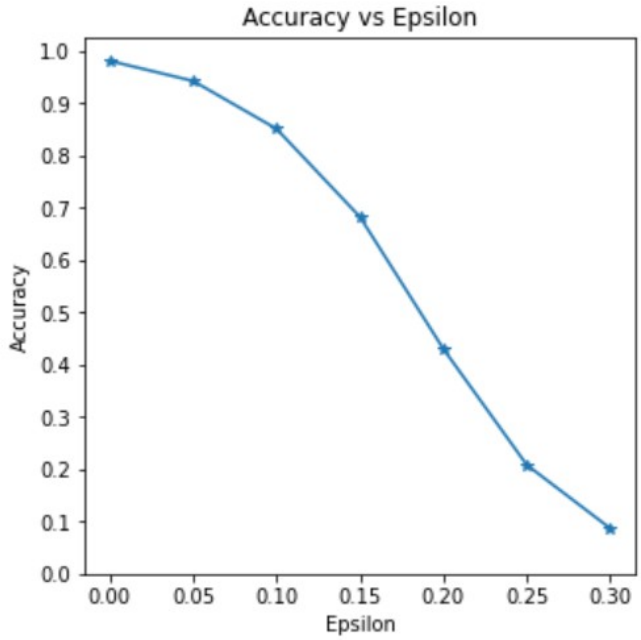
12. 调用 fgsm_attack 函数进行 FGSM 攻击，生成扰动后的数据 perturbed_data。
13. 对扰动后的数据进行再分类，得到最终预测结果 final_pred。
14. 如果最终预测与真实标签一致，增加正确分类的计数器 correct。如果 epsilon 为 0 且保存的对抗样本数量小于 5 个，则将对抗样本添加到 adv_examples 列表中。
15. 如果最终预测与真实标签不一致，将对抗样本添加到 adv_examples 列表中（数量不超过 5 个）。
16. 计算该 epsilon 下的最终准确率 final_acc。
17. 打印输出测试结果。
18. 返回最终准确率和对抗样本列表。

这段代码测试使用 FGSM 攻击后的模型在给定扰动大小 epsilon 下的准确率，并且保存一些对抗样本以供后续可视化分析。

3. 分别对 默认给出的 epsilons = [0, .05, .1, .15, .2, .25, .3]和自行修改的 epsilons 执行结果进行截图，并做简要说明。

对默认给出的 epsilons 执行的结果：

[8] ✓ 0.1s




```
# Plot several examples of adversarial samples at each epsilo
cnt = 0
plt.figure(figsize=(8,10))
```

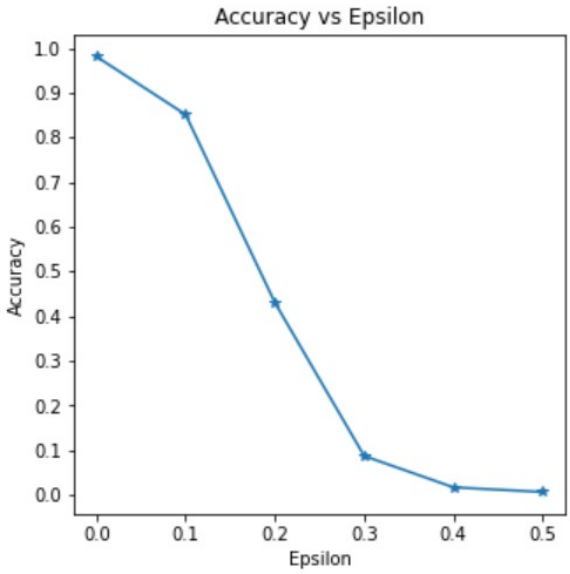
使用 `epsilons = [0, .1, .2, .3, .4, .5]`的结果：

```
# 修改数据，扩展 epsilons 的范围
epsilons = [0, .1, .2, .3, .4, .5]
```

[6] ✓ 0.0s

```
plt.figure(figsize=(5,5))
plt.plot(epsilons, accuracies, "*-")
plt.yticks(np.arange(0, 1.1, step=0.1))
plt.xticks(np.arange(0, .6, step=0.1))
plt.title("Accuracy vs Epsilon")
plt.xlabel("Epsilon")
plt.ylabel("Accuracy")
plt.show()
```

[8] ✓ 0.1s



观察两个图像，得到以下结论：

1. 当 espsilon 增大，模型正确率减小

2. espsilon 在 0.1~0.3 之间，Acc 下降最快

3. espsilon 在 >=0.3 时，Acc 下降不明显

4. 通过增大 epsilon 增强扰乱，能让正确率接近 0，效果十分明显

二、网络与信息安全实验课程的收获和建议（**必填部分**）

*（关于本学期网络与系统实验的三个部分：系统安全，网络安全和 AI 安全，请给出您对于这三部分实验的收获与体会，给出评论以及**改进的建议**。）*

1. 系统安全部分

Meltdown Attack 实验非常有趣，不过只能在实验室的老电脑上复现。

操作系统安全加固实验就只是对着 PDF 操作，对自身的提升比较有限。

2. 网络安全部分

从 SeedLabs 系列实验中了解了 PKI 公钥系统、TLS 等技术，非常有用。

3. AI 安全部分

了解了比较新颖和前沿的 AI 对抗样本攻击技术，对提升自己对 AI 安全方面的理解有很大帮助。