Week11实验

实验目标

- 在前两周的学习中，我们已经掌握了简单的投毒攻击和静态后门（可见）攻击。本周我们将在静态后门（可见）攻击的基础上进一步探索更复杂的静态后门（不可见）攻击。
- 不可见后门攻击相较可见后门攻击主要有2大区别：
  1. trigger由局部补丁的形式转变为全局噪声的形式；
  2. trigger会在训练过程中和模型一同进行优化。
- 本次实验主要是针对Lenet5模型，在MNIST数据集上进行攻击实验：
  - 指标评估
    - 验证模型在干净样本上的准确度（ACC）
    - 验证模型在加了触发器样本上的攻击成功率（ASR）
  - 可视化触发器，以及加了触发器的样本
- **Bouns**：在**CIFAR-10**上实现针对**Resnet18**的Clean-Label Attack
  - 参考《**Poison Frogs! Targeted Clean-Label Poisoning Attacks on Neural Networks**》完成Clean-Label的后门攻击
  - 在CIFAR-10测试集中随机为每一类采样10个样本作为目标样本，并在训练集中选择投毒样本完成优化，进一步微调模型最后的全连接层

实验步骤

> 本周实验包括一个不可见后门攻击的任务，和一个Bonus。

任务一：不可见后门攻击

- 根据notebook中的注释和要求，完成TODO内容

Bonus：Clean-Label Attack

- 关于bonus，我们仅提供**被测模型**与**数据预处理**部分代码，请同学自行阅读相关论文，实现关键投毒算法。以下为**几个注意点**：
  - bonus的实验设置为CIFAR-10数据集 + ResNet18模型；
  - GPU的使用能够加速投毒数据的生成与模型微调，若ModelScope的免费GPU时长仍不能满足实验要求，可换用其余免费GPU资源（如Google colab、Kaggle Notebook等）；
- **实验步骤建议：**对于一个已经训练好的Resnet-18模型，请依次做以下实现
  - 制作投毒样本
    - 在测试集中随机为每一类采样10个样本（共100个）作为目标样本，并分别随机选定目标类
    - 分别为每个目标样本选定一个训练集样本作为投毒样本，将目标样本特征作为投毒样本特征的优化目标
      - 样本特征：模型倒数第二个全连接层的输出，即最后一个全连接层的输入；

        > Tips：距离目标样本近的的投毒样本更容易优化，更容易使模型扭曲决策边界；

        - 优化投毒样本
            - 使其特征向目标样本靠拢，同时视觉上不偏离原训练集样本
                - 参考paper原文「2.2-Optimization procedure」的「Algorithm 1 Poisoning Example Generation」完成优化损失的计算
            - 使用优化后的投毒样本代替原样本，构成微调数据集
    - 微调模型
        - 在含有投毒样本的训练集上微调模型；

            > Tips：控制微调时普通样本与投毒样本之间的比例，可以改变投毒效果；

        - 冻结除了**最后一个全连接层**之外的所有层，即冻结特征提取器部分，仅对分类头微调
    - 效果评测
        - 测试并汇报在目标样本上的攻击成功率
        - 可视化优化前后的投毒样本
        - 可视化投毒前后，目标测试样本在clean model和poisoned model上的分类置信度（sigmoid(output)）变化
            - PPT中参考图像截断效果可使用工具：brokenaxes

            ```
            # 示例代码
            num_bins = 200
            bax = brokenaxes(xlims=((0, 0.1), (0.9, 1)), ylims=((0, 50), (10, 100)))
            plt.hist(confidences_1, num_bins, range=(0, 1), color="r")
            plt.hist(confidences_2, num_bins, range=(0, 1), color="b")
            ```

检查内容

- 任务一：不可见后门攻击
    1. 后门样本可视化正确
    2. 后门攻击: 3种trigger_lr设置下，最佳效果的test_acc > 95.0%、test_asr > 99.0%
    3. 随着trigger_lr的增大，ACC、ASR变化趋势正确
- Bonus：Clean-Label Attack
    1. 程序正常执行，并能理解实现细节、阐述原文方法和核心原理；
    2. 完成可视化任务（优化前后的投毒样本、目标样本在投毒前后的分类置信度变化）
    3. 汇报攻击成功率

附录

- 关于怎么获取模型倒数第一层的特征，有两种方式：
    - a) 定义钩子函数并注册`register_forward_hook`
    - b) 在原模型定义中，重写新的`forward`函数，例如添加默认参数`inspect=False`，并当`inspect=True`时返回中间特征
- Clean-Label Attack原文：[Poison Frogs! Targeted Clean-Label Poisoning Attacks on Neural Networks]
    - 请优先读懂原文的主要方法再考虑复现

    - 可参考代码（原文代码库）：https://github.com/ashafahi/inceptionv3-transferLearn-poison
- Tips：Clean-Label Attack中可能出现model.forward()单张图片的操作，请灵活使用“.unsqueeze(0)”和“.squeeze(0)”操作改变图像维度