Pseudo Label-Guided Model Inversion Attack via Conditional Generative Adversarial Network

摘要

随着深度学习技术的广泛应用，模型的隐私泄露与安全性问题日益凸显，模型反演攻击成为研究的热点。本文选择复现 PLG-MI 方法，以验证其在模型反演攻击中的有效性，并探讨其在实际应用中的可行性与局限性。在复现过程中，对原有开源代码进行了优化，包括引入 Python 的上下文管理器以防止文件描述符泄漏，以及采用 heapq 方法提升 Top-n 选择的效率。实验在 CelebA 和 FFHQ 数据集上进行了标准实验与更大分布转变实验，结果表明在保持原有攻击准确性和其他性能指标的前提下，复现后的方法在 FID 评分上较原文提升了 5% 至 20%。这些改进提升了数据加载的效率和模型训练的稳定性，证明了优化后的 PLG-MI 方法在模型反演攻击中的优越性。本文的研究为构建更加安全的深度学习模型提供了有价值的理论支持和实践指导。

关键词：模型反演攻击；白盒攻击；黑盒攻击；深度学习

1 引言

随着深度学习技术的快速发展，模型在各类应用中的表现越来越出色。然而，模型的复杂性和广泛应用也带来了隐私泄露和安全性问题。模型反演攻击作为一种能够从公开模型中恢复训练数据的方法，近年来受到了广泛关注。特别是在涉及敏感数据的场景中，模型反演攻击的威胁性尤为突出。

本文选择复现 [1] 提出的模型反演攻击方法，旨在验证其方法的有效性，并探讨其在实际应用中的可行性和局限性。通过对该领域的深入研究，本文希望为构建更加安全的深度学习模型提供理论支持和实践指导。

2 相关工作

本部分对模型反演攻击领域的相关研究进行了简要的分类概述，包括白盒攻击和黑盒攻击两大类。

2.1 白盒攻击

白盒攻击是指攻击者对目标模型拥有完全的访问权限，包括模型的架构、参数和训练数据等信息。在这种情况下，攻击者可以利用这些详细信息进行高效的反演攻击。例如 [2]，该文提出了一种基于变分推断和生成对抗网络（GAN）的优化方法，通过变分推断，能够有效地通过优化后验分布，增强攻击的准确性，减少梯度信息的局限性。再例如 [3](方法如图1所示)，该文提出了一种新颖的攻击方法（GMI），能够以高成功率反演深度神经网络。这种方法并非从零开始重建私有训练数据，而是利用部分公开信息），通过生成对抗网络（GAN）学习一个分布式的先验信息，并用其引导反演过程。白盒攻击通常具有较高的攻击成功率，但在实际应用中，由于获取完整模型信息的难度较大，其应用场景相对有限。

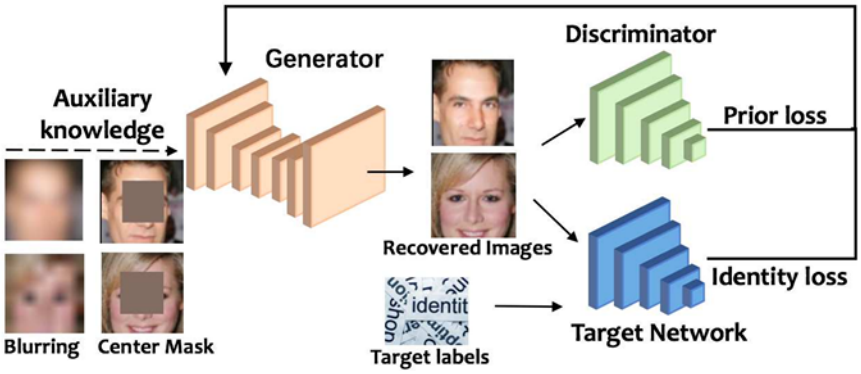

图 1. GMI 方法的框架图

2.2 黑盒攻击

相比之下，黑盒攻击假设攻击者只能访问模型的输入和输出，无法获取模型的内部结构和参数信息。尽管如此，黑盒攻击仍然能够通过观察模型的输出行为，逐步推断出训练数据的敏感信息。例如 [4](方法如图2所示), 提出了一种新的方法——基于强化学习的黑盒模型反演攻击（RLB-MI）。将潜在空间的探索问题建模为马尔可夫决策过程（MDP），并通过强化学习对生成图像的置信度分数提供奖励，指导智能体在潜在空间中进行有效的搜索。黑盒攻击的优势在于其更广泛的适用性，但由于缺乏内部信息，攻击过程通常更加复杂且耗时。

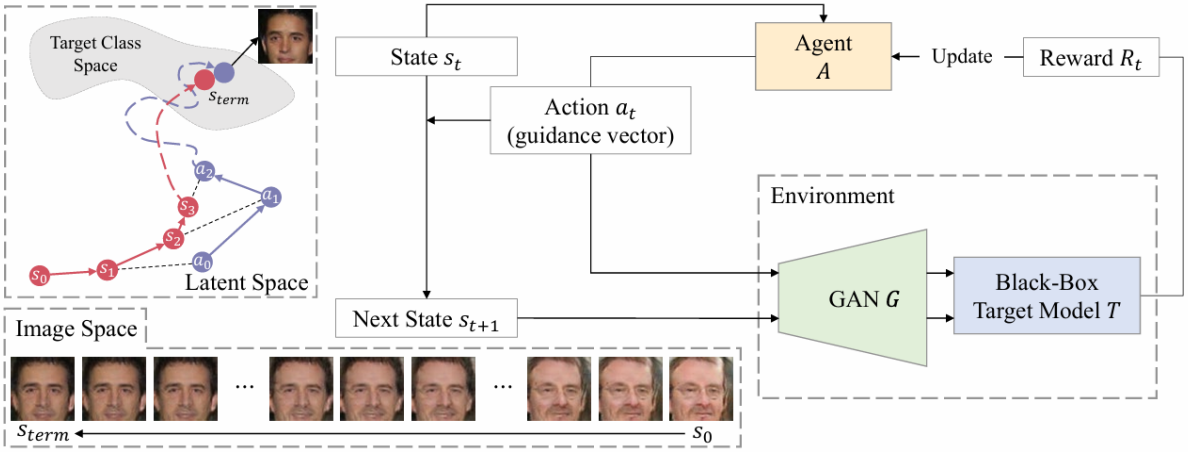

图 2. RLB-MI 方法的框架图

3 本文方法

3.1 Top-n 方法

Top-n 选择策略在 PLG-MI 方法中用于从公共数据集中筛选出最符合目标类别特征的样本并分配伪标签。具体步骤包括将公共数据输入目标模型，获取每个样本的预测置信度，然后针对每个类别选择预测置信度最高的前 n 个样本作为伪标签数据。这一策略通过确保选中的样本与目标类别高度相关，提高了伪标签的准确性，减少了训练中的噪音，从而帮助条件生成对抗网络（cGAN）更有效地学习目标类别的特征分布。选择合适的 n 值能够在伪标签的准确性与样本多样性之间取得平衡，确保生成器在模型反演阶段生成高质量的重构图像。Top-n 方法的示意图如图 (3) 所示：

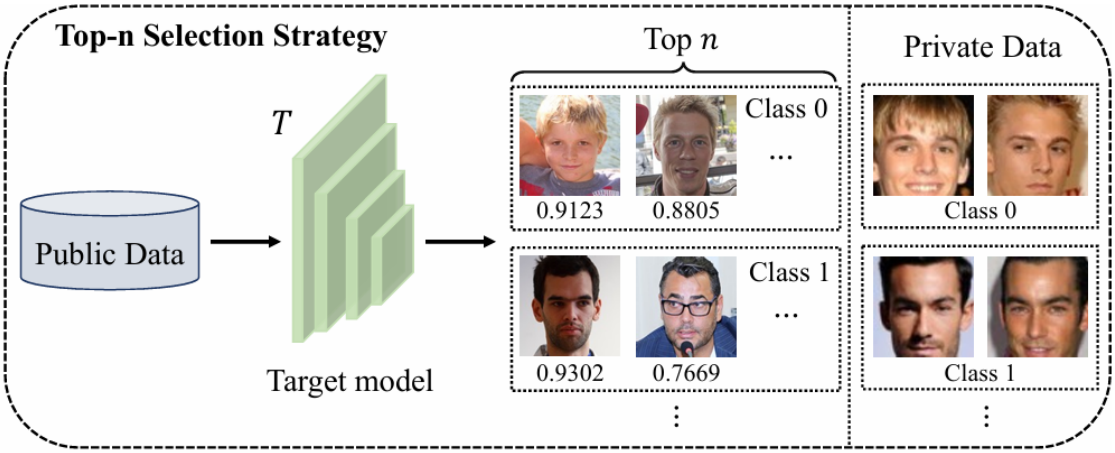

图 3. Top-n 选择方法

3.2 攻击流程

攻击流程的框架图如图 (4) 所示。

1. 利用公共数据集输入目标模型，获取每个样本的预测置信度，并采用 top-n 策略为每个类别选择预测置信度最高的 n 个样本，赋予其伪标签。
2. 使用这些带有伪标签的公共数据训练条件生成对抗网络（cGAN），使生成器能够学习特定类别的特征分布。
3. 训练带类别条件的 cGAN，确保生成器专注于目标类别的特征子空间。
4. 在生成器的潜变量空间中，通过引入 max-margin 损失，优化生成的图像以最大化目标模型对特定类别的响应，从而重构私有数据。

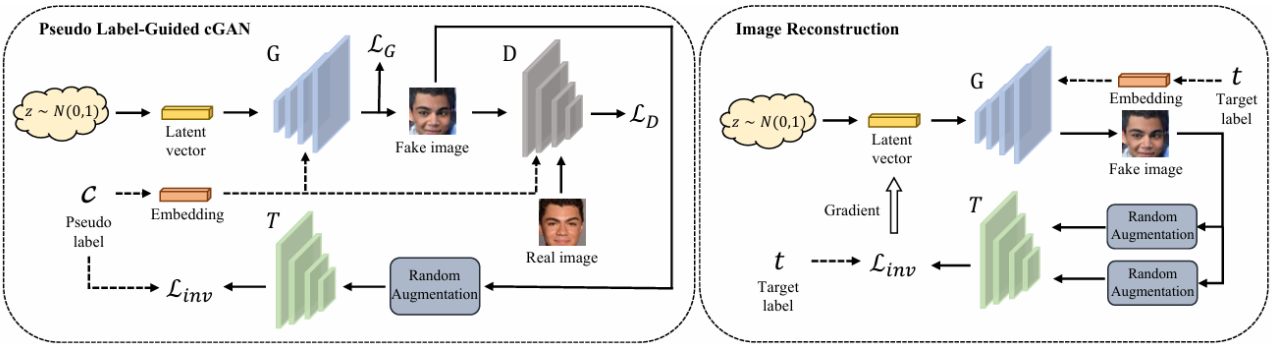

图 4. 攻击流程的框架

3.3 损失函数

本文中主要采用了以下两种损失函数：

3.3.1 条件生成对抗网络（cGAN）损失

PLG-MI 方法中首先训练一个条件生成对抗网络（cGAN）以学习目标类别的特征分布。cGAN 的损失函数由生成器（Generator）损失和判别器（Discriminator）损失组成，具体定义如下：

$$\mathcal{L}_{\text{cGAN}} = \mathbb{E}_{x,y}\left[\log D(x|y)\right] + \mathbb{E}_{z,y}\left[\log\left(1 - D(G(z|y)|y)\right)\right] \tag{1}$$

其中：

- $G(z|y)$ 表示生成器接受潜在变量 z 和类别条件 y 后生成的假图像。
- $D(x|y)$ 表示判别器接受图像 x 和类别条件 y 后判断其真实性的概率。
- 第一项 $\mathbb{E}_{x,y}\left[\log D(x|y)\right]$ 是判别器对真实样本的判别损失。
- 第二项 $\mathbb{E}_{z,y}\left[\log\left(1 - D(G(z|y)|y)\right)\right]$ 是判别器对生成样本的判别损失。

通过优化 $\mathcal{L}_{\text{cGAN}}$，生成器学会生成与条件 y 相对应的逼真图像，而判别器则提高区分真实图像与生成图像的能力。

3.3.2 最大间隔（Max-Margin）损失

在模型反演的第二阶段，PLG-MI 方法采用了最大间隔（Max-Margin）损失来替代传统的交叉熵（Cross-Entropy, CE）损失，以解决梯度消失的问题并提升反演图像的质量。最大间隔损失的定义如下：

$$\mathcal{L}_{\text{Max-Margin}} = \max(0, m + f_y(x) - \max_{k \neq y} f_k(x)) \tag{2}$$

其中：

- $f_y(x)$ 表示目标类别 y 的 logit 值，即目标模型对类别 y 的预测得分。

- $\max_{k \neq y} f_k(x)$ 表示除目标类别 y 外，其他类别的最大 logit 值。
- m 是一个预设的间隔超参数，用于确保目标类别的得分显著高于其他类别。

该损失函数的目标是最大化目标类别 y 的得分与其他类别得分之间的间隔。当 $f_y(x)$ 与 $\max_{k \neq y} f_k(x)$ 之间的差距超过 m 时，损失为零；否则，损失将推动生成器增加目标类别的得分并减少其他类别的得分。与交叉熵损失不同，最大间隔损失不会因为 softmax 函数的饱和而导致梯度消失，从而保持了有效的梯度信号，促进生成器生成更能激活目标模型的高质量图像。

3.4 FID（Fréchet Inception Distance）

FID（Fréchet Inception Distance）是一种广泛用于评估生成模型（如生成对抗网络，GAN）生成图像质量的指标。它通过比较生成图像与真实图像在特征空间中的分布差异，量化生成图像的质量和多样性。FID 在生成模型的研究和应用中具有重要意义，因为它提供了一种客观且敏感的评估方法。FID 基于 Fréchet 距离（也称为 Wasserstein-2 距离），用于衡量两个多变量高斯分布之间的差异。在计算 FID 时，首先使用预训练的 Inception 网络（通常是 Inception v3）提取生成图像和真实图像的高层特征，然后分别计算这两组特征的均值和协方差矩阵。最后，FID 通过以下公式计算：

$$\text{FID} = \|\mu_r - \mu_g\|^2 + \text{Tr}\left(\Sigma_r + \Sigma_g - 2(\Sigma_r \Sigma_g)^{1/2}\right) \tag{3}$$

其中：

- μ_r 和 Σ_r 分别是真实图像特征的均值和协方差矩阵。
- μ_g 和 Σ_g 分别是生成图像特征的均值和协方差矩阵。
- Tr 表示矩阵的迹。

4 复现细节

4.1 与已有开源代码对比

文章的代码开源在https://github.com/LetheSec/PLG-MI-Attack. 但是代码并不能直接运行，还作出了以下修改：

- 对于“打开文件的数量超过了操作系统的设置”的问题，采用上下文管理器（with 语句），确保图像在使用后被关闭
- 对于“Top-n 选择效率较低”的问题，使用 python 中 heapq 方法，提升 Top-n 选择的效率

在未使用 with 语句之前，程序可能由于文件描述符泄漏（即文件未及时关闭），导致部分图像文件未被正确读取或加载不完整。这会导致数据集中存在损坏或不完整的图像，这些图像在训练过程中可能引入噪音，影响模型的学习效果。加入 with 语句后（如图5所示），可

以确保所有图像数据都被完整且正确地加载，避免了由于数据不完整导致的训练噪音，提高了模型训练的质量，从而提升了生成图像的 FID 评分。

```
num_classes = len([lists for lists in os.listdir(
    self.path) if os.path.isdir(os.path.join(self.path, lists))])

for idx in range(num_classes):
    class_path = os.path.join(self.path, str(idx))
    for _, _, files in os.walk(class_path):
        for img_name in files:
            image_path = os.path.join(class_path, img_name)
            with Image.open(image_path) as image:
                if args.data_name == 'facescrub':
                    if image.size != (64, 64):
                        image = image.resize( size: (64, 64), Image.ANTIALIAS)
                self.images.append((image.copy(), idx))
```

图 5. 加入 with 语句后的部分代码

4.2 实验环境搭建

实验环境主要基于 Python 3.9。硬件方面，实验在配备 NVIDIA Tesla P100 16GB 显卡的服务器上进行，以确保训练过程的高效性。所有依赖库的版本信息如下：

- pytorch 1.9.0+cu102
- numpy 1.23.2
- torchvision 0.10.0+cu102
- opencv-python 4.10.0

4.3 创新点

通过采用 Python 的 with 语句优化图像文件的管理，有效防止了文件描述符泄漏问题。这一改进不仅解决了“open too many files”的错误，还提升了数据加载的效率和系统资源的管理能力，从而提高了模型训练的稳定性和生成图像的质量，最终实现了比原文更优的 FID 评分。

5 实验结果分析

本部分对复现实验的结果进行了详细分析。实验使用的数据集包括 CelebA 和 FFHQ。

表 1. 复现后在不同模型下的实验结果

	VGG16	ResNet-152	Face.evoLVe
Attack Acc	**0.98**	**1.00**	**0.99**
Top-5 Attack Acc	**1.00**	**1.00**	**1.00**
KNN Dist	**1126.54**	**1020.05**	**1102.72**
FID	**15.95**	**21.69**	**20.93**

5.1 标准实验

标准实验指的是公有数据集和私有数据集都为 CelebA 数据集，复现的实验结果如表1所示，原文的实验结果如图6所示：

	VGG16			ResNet-152			Face.evoLVe		
	GMI	**KED-MI**	**Ours**	**GMI**	**KED-MI**	**Ours**	**GMI**	**KED-MI**	**Ours**
Attack Acc ↑	.21±.0028	.63±.0018	**.97±.0001**	.31±.0035	.74±.0028	**1.±.0000**	.29±.0030	.74±.0013	**.99±.0001**
Top-5 Attack Acc ↑	.42±.0021	.87±.0015	**1.±.0000**	.55±.0045	.93±.0006	**1.±.0000**	.54±.0040	.94±.0009	**1.±.0000**
KNN Dist ↓	1712.57	1391.52	**1120.61**	1630.25	1323.16	**1026.71**	1638.94	1310.15	**1103.03**
FID ↓	42.86	30.92	**18.63**	42.50	26.23	**23.22**	41.53	27.92	**26.75**

图 6. 原文在不同模型下的实验结果

5.2 更大分布转变实验

该实验指的是公有数据集变为 ffhq，私有数据集仍为 CelebA，复现的实验结果如表2所示，原文的实验结果如图7所示：

表 2. 复现后不同模型在 FFHQ 到 CelebA 数据集上的性能对比

$ffhq \rightarrow CelebA$	VGG16	ResNet-152	Face.evoLVe
Attack Acc	**0.88**	**0.97**	**0.95**
Top-5 Attack Acc	**0.97**	**1.00**	**0.98**
KNN Dist	**1300.61**	**1147.84**	**1262.69**
FID	**20.64**	**21.56**	**17.89**

		FFHQ → CelebA			
		Attack Acc ↑	**Attack Acc 5 ↑**	**KNN Dist ↓**	**FID↓**
VGG16	**GMI**	.11±.0009	.27±.0048	1771.34	57.05
	KED-MI	.34±.0026	.62±.0015	1555.57	49.51
	Ours	**.89±.0006**	**.97±.0002**	**1284.16**	**27.32**
Face.evoLVe	**GMI**	.13±.0009	.31±.0028	1739.88	56.66
	KED-MI	.47±.0021	.74±.0013	1489.67	44.48
	Ours	**.95±.0004**	**.99±.0001**	**1241.41**	**25.57**
ResNet-152	**GMI**	.17±.0026	.37±.0030	1687.82	47.11
	KED-MI	.74±.0028	.93±.0006	1323.16	26.23
	Ours	**1.±.0000**	**1.±.0000**	**1026.71**	**23.22**

图 7. 原文在不同模型下从 FFHQ 到 CelebA 数据集上的实验结果

我们可以清楚地观察到复现以后的 FID 指标, 不管是标准实验还是更大分布转变实验, 在其他指标和原文区别不大的情况下，均要低 **5%** 至 **20%** 不等。由此可以看出修改以后的代码运行效果更好。

6 总结与展望

虽然加入了 with 语句，提升了 FID 效果，然而，本文的研究也存在一定的局限性。实验仅在两个特定的数据集上进行，未能涵盖更多类型的数据集和模型架构，限制了结果的泛化性。此外，本文主要聚焦于攻击方法的复现与优化，尚未深入探讨相应的防御机制，这在实际应用中具有重要意义。

未来的研究可以从以下几个方面展开：

1. **扩展实验范围**：在更多类型的数据集和不同架构的模型上验证优化方法的有效性，提升研究结果的普适性。
2. **增强攻击策略**：探索结合其他优化技术或先进的生成模型，以进一步提升模型反演攻击的成功率和生成图像的质量。
3. **防御机制研究**：在模型反演攻击的基础上，研究相应的防御策略，如差分隐私、模型蒸馏等方法，以构建更加安全的深度学习模型。

参考文献

[1] Xiaojian Yuan, Kejiang Chen, Jie Zhang, Weiming Zhang, Nenghai Yu, and Yang Zhang. Pseudo label-guided model inversion attack via conditional generative adversarial network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 3349–3357, 2023.

[2] Kuan-Chieh Wang, YAN FU, Ke Li, Ashish Khisti, Richard Zemel, and Alireza Makhzani. Variational model inversion attacks. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 9706–9719. Curran Associates, Inc., 2021.

[3] Yuheng Zhang, Ruoxi Jia, Hengzhi Pei, Wenxiao Wang, Bo Li, and Dawn Song. The secret revealer: Generative model-inversion attacks against deep neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[4] Gyojin Han, Jaehyun Choi, Haeil Lee, and Junmo Kim. Reinforcement learning-based black-box model inversion attacks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 20504–20513, June 2023.