TransTroj: Transferable Backdoor Attacks to Pre-trained Models via Embedding Indistinguishability

> 2024 arXiv
> https://arxiv.org/abs/2401.15883

已有的针对下游任务不可知的后门攻击方法存在下调过程灾难性遗忘、不能完全任务无关（若微调任务没被覆盖到后门就失效）的问题，提出的 TransTroj 通过系统分析投毒样本和目标类别样本的 embedding 空间的不可分辨性来保证 `functionality-preserving、durable、task-agnostic`。
将 embedding 的不可分辨性（indistinguishability）拆解成 pre-/post-indistinguishability，分别代表投毒样本和目标样本的 clean PTM/后门PTM 上 embedding 空间的 indistinguishability。为此指制定两个阶段的优化目标：

1. 通过聚合公开可获取目标样本的 embedding来创建参考 embedding 并优化一个 pervasive trigger 来增强投毒样本和参考 embedding 之间的 pre-indistinguishability；
2. 在制作的投毒数据集上优化受害PTM以增强 post-indistinguishability。

pre-indistinguishability 就是让投毒样本和目标类样本在进入模型前，从 embedding 开始就相似；post-indistinguishability 是在模型训练能产生后门。

§3 问题定义

系统模型：

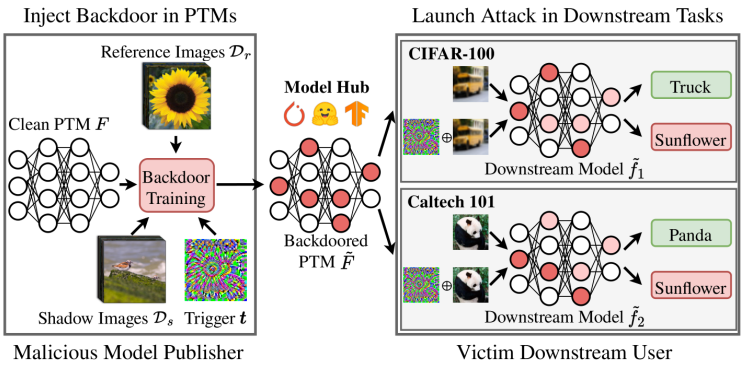


Figure 2. Illustration of transferable backdoor attacks. The adversary injects backdoor into a clean PTM and launches attack when the backdoored PTM is leveraged to fine-tune downstream tasks.

- Model Publisher (MP) 先用自己的训练集训练/微调一个PTM F，然后将梯度 θ^F 发布在公共平台上，Downstream User (DU) 下载公共PTM并用于自己的下游任务；
- 针对分类的下游任务，对应的模型 f 通常在 F 上附加一个线性分类头来构建，此时由DU自己微调的优化目标为

$$\min_{\theta^F,\theta^h}\{E_{(x,y)\in D_t}L(f(x),y)\}$$

威胁模型：

- 攻击者目标：模型发布者是恶意的，可以发布恶意PTM $\tilde{F}$。如图2，攻击者初始化干净PTM F，选目标嘞 y_t 后优化对应 trigger t，受害者DU下载 $\tilde{F}$ 并微调后用包含 trigger t 的样本时后门要能成果

触发。

- 可转移后门满足以下目标：
  - Functionality-preserving：$\tilde{F}$ 保持其原始功能，由 $\tilde{F}$ 构建的下游模型 $\tilde{f}$ 的精度要和干净模型一样；
  - Durable：微调可能会导致模型的灾难性遗忘，需避免后门在微调过程中遗忘；
  - Task-agnostic：可转移后门应该对任何任务都有效，而不是针对某特定任务，当目标类 y_t 包含在下游任务中时，$\tilde{f}$ 应该对任何包含 trigger t 的输入 x 都预测为 y_t，即 $y_t = \tilde{f}(x \oplus t)$。
- 攻击者能力：①对PTM完全掌握，以及一组可访问的图像作为 shadow dataset D_s；同时对每个目标类还需要一小组参考图像 D_r ；②对下游任务无先验知识

> 这里 D_r 多留意一下后面有什么用，既然都任务无关了还要特别为目标类准备数据集，怀疑是不是还是没脱离假设，仍需要训练恶意PTM时就覆盖到下游任务

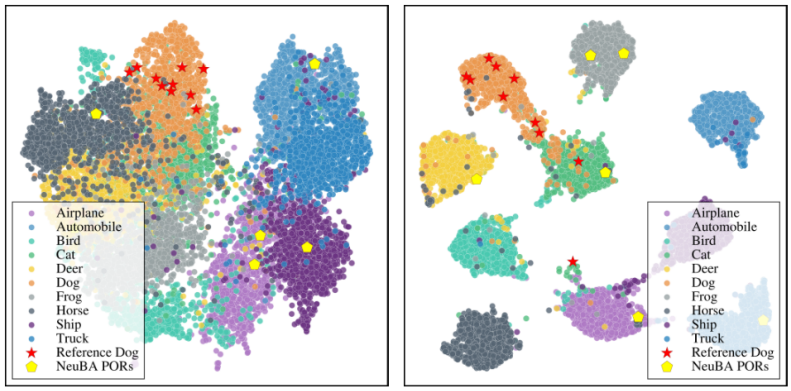
§4 方法

前期：

- 两个现象：
  1. 以前的攻击方法用手工制作的后门 trigger，比如在输入图像的右下角加一个 patch，但是由于不知道下游数据 trigger 的模式，这种方式容易被模型遗忘。作者的观点是 trigger 与目标类有语义相关性，那么目标类的样本在下游数据中就可以提供持续的后门。

     > 这里存疑

  2. 一些将 trigger 和 POR（预定义输出表示）绑定的攻击，即使同时嵌入了多个POR到PTM中，由于缺乏先验知识，也不一定包含到目标类别，比如图中对目标类是 dog 嵌入的多个POR就没覆盖到。



(a) Pre-trained Model (b) Downstream Model

Figure 3. Visualization of dimension-reduced embeddings [28] of CIFAR-10 dataset extracted by ResNet-18. The reference dog images, denoted as red stars, are downloaded from the Internet.

- 设计关键在可转移后门攻击应当使有毒输入和干净输入在 embedding 空间上难以区分，比如PTM为投毒样本生成的 embedding 和 dog 图像的相似，下游模型就会错误的将有毒样本错误分类成 dog。攻击流程如下：

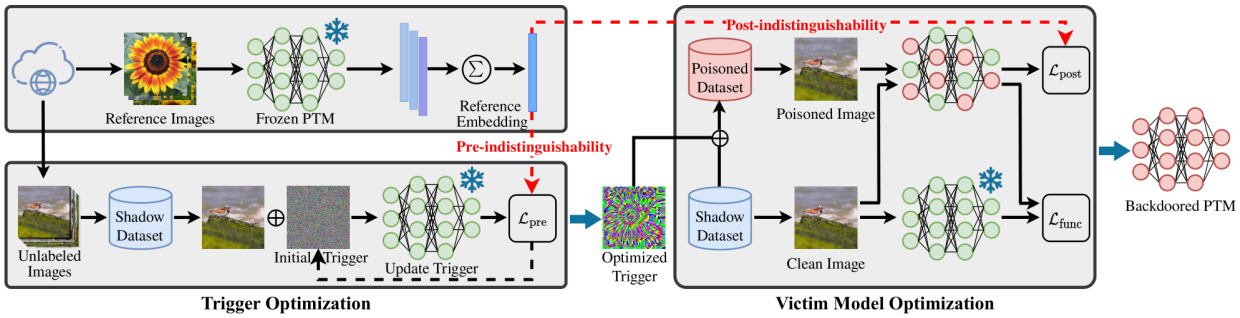


Figure 4. The pipeline of TransTroj. We first optimize a trigger to make the poisoned images similar to the reference images, *i.e.*, pre-indistinguishability. Then, we optimize the victim PTM such that the poisoned embeddings and reference embeddings cannot be distinguished, *i.e.*, post-indistinguishability.

由于无法访问下游任务的目标类别图像，攻击者可以从网上获取参考图像来估算参考 embedding 输入，通过模型优化和 trigger 优化增强投毒图像和干净图像的相似度。

> 这里保证 embedding 相似应该是借助微调过程，微调对干净样本会重复训练使模型记住，从而对相同 embedding 的后门也能被记住。

- Transferable Backdoor Attacks：有后门的下游模型 $\tilde{f}$ 在干净图像 x 上表现正常，但是有 trigger t 的图像上会将其预测为目标类别 y_t，因此优化问题为

$$\max_{\theta_{\tilde{f}}} \sum_{(x,y)\in D_t} [\mathrm{I}\left(\tilde{f}(x)=y\right)+\mathrm{I}\left(\tilde{f}(x\oplus t)=y_t\right)]$$

但是因为缺乏下游知识，无法指导 f 和 D_t，因此将上述优化问题转换：

1. 错误分类投毒样本的目标 → 使得投毒样本和干净样本的 embedding 空间难以区分，就是让PTM对投毒样本预测的 embedding 和目标类相似；
2. 下游任务数据集的访问 → 公开未标记的 shadow dataset 和参考图像，从互联网上下载的的图像是可以替代真实的目标类别图像的。

Pre- and Post-indistinguishability:

- Definition 1 (Pre-indistinguishability)：有毒样本 $x\oplus t$、目标类的干净样本 x_t，由干净PTM F 提取 $x\oplus t$ 和 x_t 的 embedding，如果 $d(\cdot,\cdot)$ 计算超过相似性阈值 ϵ_1，就认为二者是pre-indistinguishable 的

$$d(F(x\oplus t),F(x_t))>\epsilon_1$$

d可以用余弦相似度等。

  - Pre-indistinguishability 通过投毒样本与干净样本在 embedding 上的相似性来保证了后门的持久性，最直接的方式是优化触发器以实现这种相似

$$\max_{t}\frac{1}{|D_s|\cdot|D_r|}\sum_{x_s\in D_s}\sum_{x_r\in D_r}d(F(x_s\oplus t),F(x_r))$$

  reference images $|D_r|$ 不需要太多，有少于10个的目标类的参考图像就行

- Definition 2 (Post-indistinguishability)：和 pre-indistinguishability 相似的定义

$$d(\tilde{F}(x\oplus t),\tilde{F}(x_t))>\epsilon_2$$

  - 进一步强化投毒PTM在投毒样本和正常样本之间的相似性，pre- 是优化 trigger，post- 是优化模型

$$\max_{\theta^{\tilde{F}}}\frac{1}{|D_s|\cdot|D_r|}\sum_{x_s\in D_s}\sum_{x_r\in D_r}d(\tilde{F}(x_s\oplus t),\tilde{F}(x_r))$$

两个阶段优化：

- Trigger optimization：现在常用的 patch-like trigger 是离散的像素值，不利于优化，本文解决方案是用全局 perturbations 作为 trigger，利用每个像素的轻微 perturbations

$$x \oplus t = clip(x + t, 0, 255)$$

$clip(\cdot, 0, 255)$ 限制了像素的有效值，x 和 t 应具有相同的形状，通过限制 $||t||_\infty \leq \xi$ 确保 trigger 的扰动不会太大，保持稳定性和有效性。因此用 loss 来量化优化目标

$$L_{pre} = \frac{1}{|D_s|} \sum_{x \in D_s} d(F(x \oplus t), r)$$

r 是参考 embedding，是所有参考图像 embedding 的平均值 $r = \frac{1}{n} \sum_{i=1}^{|D_r|} F(x_{r_i})$，因此优化目标变为

$$\arg\min_t L_{pre},\ ||t||_\infty \leq \xi$$

- Victim PTM optimization：模型有后门同时保持干净图像的精度

$$L_{post} = \frac{1}{|D_s|} \sum_{x \in D_s} d(\tilde{F}(x \oplus t), r)$$

$$L_{func} = \frac{1}{|D_s|} \sum_{x \in D_s} d(\tilde{F}(x), F(x))$$

优化目标

$$\arg\min_{\theta_{\tilde{F}}} L = L_{post} + \lambda L_{func}$$

§5 实验

实验设置：

- 预训练模型和数据集：ResNet、VGG、ViT、CLIP 四个 PTM，在 ImageNet1K 数据集上训练 ResNet、VCG（CNN）、ViT（Transformer），CLIP 包含图像文本对但本文后门针对图像 encoder 因此使用 PyTotch、Hugging Face 提供的预训练权重。
- 下游任务：6个下游任务， CIFAR-10、CIFAR-100、GTSRB、Caltech 101、Caltech 256、Oxford-IIIT Pet。
- 评估指标：Clean Accuracy (CA)-干净下游模型的分类精度、 Attack Success Rate (ASR)-投毒样本被分类为目标类、Backdoored Accuracy (BA)-后门下游模型在良性任务上的精度，通过对比CA、BA可以判断PTM是否包含原始功能。
- 一些设置：D_s 有 50,000 图像，$||t||_\infty \leq \xi = 10$，$\lambda = 10$，微调学习率 $1e-4$、$1e-5$，微调20 epoch。
- Baseline：BadEncoder、NeuBA

Attack Effectiveness：

- Functionality-preserving：

Method	Model	CIFAR-10			CIFAR-100			GTSRB			Caltech 101			Caltech 256			Oxford-IIIT Pet		
		CA	BA	ASR	CA	BA	ASR	CA	BA	ASR	CA	BA	ASR	CA	BA	ASR	CA	BA	ASR
BadEncoder [19]	ResNet-18	95.45	95.20	9.81	80.11	79.48	1.12	98.54	98.84	5.75	96.14	95.68	1.04	82.03	81.49	26.87	88.09	88.50	81.28
	VGG-11	91.52	91.18	9.64	70.67	70.93	3.81	99.08	99.02	5.71	93.09	93.95	58.99	74.64	74.50	33.59	87.03	83.73	75.14
	ViT-B/16	98.26	97.85	0.09	85.78	86.33	0.41	98.65	98.87	0.25	96.72	96.03	0.06	85.47	85.70	0.07	92.75	92.64	0.11
NeuBA [44]	ResNet-18	92.07	91.02	13.74	73.33	71.67	4.62	95.60	95.02	7.71	88.13	85.54	9.85	62.56	58.38	2.92	60.43	48.98	4.47
	VGG-11	91.93	90.94	57.17	70.56	70.88	49.35	96.28	95.86	53.29	89.69	89.69	63.88	62.22	59.74	48.69	69.58	69.69	60.53
	ViT-B/16	95.95	96.18	81.95	84.49	84.65	78.19	95.74	95.81	67.91	88.88	88.94	64.92	75.72	75.37	56.30	73.48	74.05	64.98
Ours	ResNet-18	95.45	95.41	**100.0**	80.11	80.25	**100.0**	98.54	98.80	**100.0**	96.14	95.79	**98.68**	82.03	81.27	**98.79**	88.09	88.42	**99.73**
	VGG-11	91.52	92.00	**99.51**	70.67	71.49	**100.0**	99.08	99.13	**94.03**	93.09	95.45	**93.95**	74.64	74.37	**91.47**	87.03	83.46	**99.07**
	ViT-B/16	98.26	97.91	**100.0**	85.78	86.03	**100.0**	98.65	98.95	**100.0**	96.72	96.08	**99.36**	85.47	85.55	**99.80**	92.75	89.26	**100.0**

Table 1. Comparison of attack performance on different PTMs and downstream tasks. The highest ASR of each dataset is in boldface.

90%+的ASR → 后门成功率高；CA与BA差距在1%内 → 正常功能保持完好；

- durable：

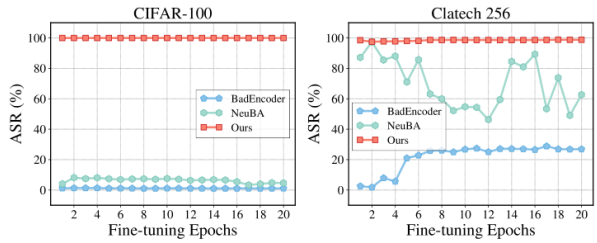


Figure 1. Attack success rates when fine-tuning the backdoored PTM for different downstream tasks. SOTA methods (*i.e.*, BadEncoder [19] and NeuBA [44]) achieve only limited performance across a subset of downstream tasks. In contrast, our method achieves a high attack success rate across various downstream tasks and remains stable during the fine-tuning process.

> 这BadEncoder怎么还涨了

- 任务无关：选了向日葵、豹子、袋鼠三个目标类，在三个不同数据集代表的下游任务上测试

Target class	Downstream dataset	CA	BA	ASR
Sunflower	CIFAR-100	80.11	80.25↑ 0.14	100.0
	Caltech 101	96.14	95.79↓ 0.35	98.68
	Caltech 256	82.03	81.27↓ 0.76	98.79
Leopard	CIFAR-100	80.11	80.17↑ 0.06	100.0
	Caltech 101	96.14	95.85↓ 0.29	97.24
	Caltech 256	82.03	81.29↓ 0.74	99.04
Kangaroo	CIFAR-100	80.11	79.71↓ 0.40	99.99
	Caltech 101	96.14	95.68↓ 0.46	98.21
	Caltech 256	82.03	81.32↓ 0.71	97.39

Table 2. Results of simultaneously attacking three target downstream datasets through one target class. ↑ 0.14 indicates that the backdoor accuracy is 0.14% higher than the clean accuracy.

实验结果仍然很高，只要目标类包含在下游任务中就可以做到任务无关（因为要与目标embedding绑定来防止灾难遗忘）

Multi-target Backdoor Attacks：

- 同时存在多个目标类进行攻击，也就是目标是一组类 $(y_1, y_2, \ldots, y_n)$ 和一组 trigger $(t_1, t_2, \ldots, t_n)$ 对应绑定，测试了五个下游任务

Downstream dataset	CA	BA	ASR
$\text{CIFAR-100}_{\text{Leopard}}$	80.11	79.75↓ 0.36	99.89
$\text{GTSRB}_{\text{Yield sign}}$	98.54	98.80↑ 0.26	99.98
$\text{Caltech 101}_{\text{Sunflower}}$	96.14	95.45↓ 0.69	98.21
$\text{Caltech 256}_{\text{Dog}}$	82.03	81.04↓ 0.99	99.27
$\text{Oxford-IIIT Pet}_{\text{Samoyed}}$	88.09	87.79↓ 0.30	99.51

Table 3. Results of attacking 5 target classes simultaneously.

Sensitivity Analysis：

- trigger 范数 ξ 、shadow dataset 大小的影响，使用 ResNet-18 作PTM、Caltech 256 作下游任务

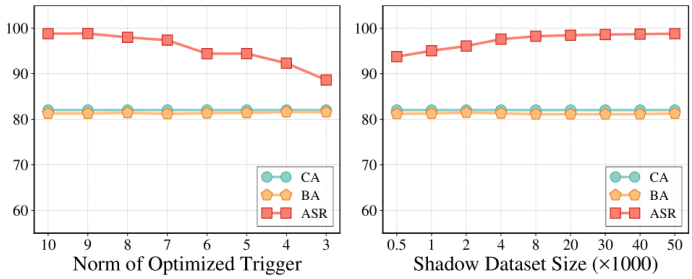


Figure 6. The impact of the optimized trigger infinity norm ξ (left) and the shadow dataset size $|\mathcal{D}_s|$ (right).

范数过小会导致trigger不能有效插入使得两个embedding相似

- 消融实验：loss terms、trigger 模式、参考图像的影响

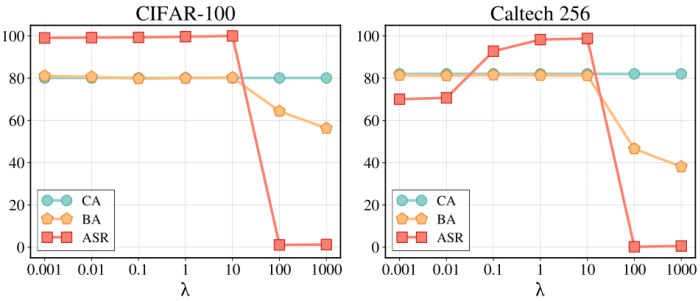


Figure 13. The impact of the λ.

Tigger pattern	Downstream dataset	CA	BA	ASR
Patch	CIFAR-100	80.11	79.85↓ 0.26	68.84
	Caltech 101	96.14	95.74↓ 0.40	10.94
	Caltech 256	82.03	81.34↓ 0.69	6.29
SIG	CIFAR-100	80.11	79.52↓ 0.59	1.43
	Caltech 101	96.14	95.74↓ 0.40	90.61
	Caltech 256	82.03	81.79↓ 0.24	74.69
Random	CIFAR-100	80.11	79.44↓ 0.67	1.54
	Caltech 101	96.14	95.51↓ 0.37	19.64
	Caltech 256	82.03	81.36↓ 0.67	0.84
Optimized	CIFAR-100	80.11	80.25↑ 0.14	100.0
	Caltech 101	96.14	95.79↓ 0.35	98.68
	Caltech 256	82.03	81.27↓ 0.76	98.79

patch的效果比较差（附录里有添加trigger的可视化图片）

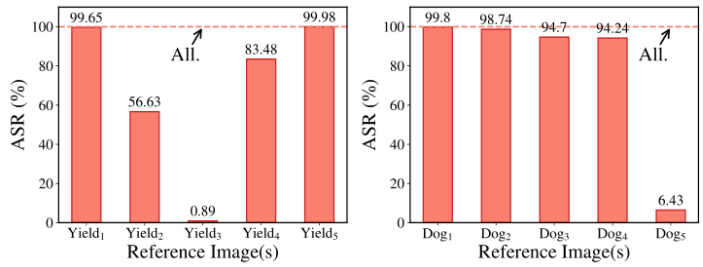


Figure 15. Results of attacking “Yield” (left) and “Dog” (right) using a single reference image. “All” refers to the attack success rate when using the average embedding of 10 reference images.

一些参考图片会导致后门失败，因为不是所有模型都能识别这张图片，所以采用多张参考图片而非单一图片

Cause Analysis：后门成功关键在投毒样本与目标类样本之间的不可区分，实验也表明后门PTM的注意力主要集中在图像中间区域，也就是说图片中间区域对模型影响较大，因此下游模型忽略了有毒输入的特定内容从而预测成目标类。

> 个人理解：正常训练干净图像注意力集中在图像主体位置，在主体位置之外影响较小，因此在这个地方添加 trigger 不会干扰到其他类别的正常预测，因为这部分像素影响小对权重改变小，但是加了 trigger 后能更改模型的注意力范围（原文图8能看出来），直接诱导成目标类，这也算是相较于小 patch 全像素做修改投毒的优势。

鲁棒性：

- ResNet-18 作为 PTM，Oxford-IIIT Pet 作为下游任务。考虑的后门防御手段有 Re-initialization 重新初始化PTM的最后几个卷积层，Fine-pruning 关闭在干净输入上处于休眠状态的神经元，被屏蔽的通道比例由修剪率控制

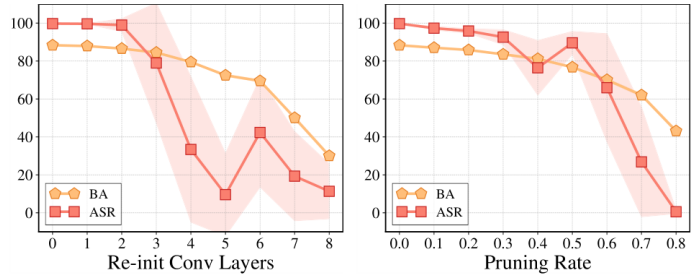


Figure 9. The average (compiled from 100 trials) attack success rate and backdoor accuracy after re-initialization and fine-pruning.

模型防御也不会让自己模型精度太低，所以后门都能在模型本身精度下降前保持稳定就行。