Knockoff Nets: Stealing Functionality of Black-Box Models

- 本文主要研究的场景在于输入一张图像, 然后输出预测的black box API. 相比于之前的工作, 本文中的攻击者缺少关于victim model的train/test数据集的知识, 其内部结构, 以及输出的语义信息等
- 本文提出一种两步骤的方法. 1)是通过输出图像对victim model进行请求, 以获得对应的预测. 2)使用之前构造的数据集训练一个本文提出的knockoff进行模型抽取
- 本文一些观察 : 1)使用随机不同分布的图片去对victim model进行请求, 而不是使用其训练数据, 可以在knockoff取得很好的效果. 2)knockoff不用与victim相同的架构是可能的. 3)在某些setting中, 本文使用的reinforcement learning方法可以额外的提高请求的效率, 并提供性能的提升
- 本文是面向纯粹偷取复杂模型的functionality
- **提到knowledge transfer / knowledge distillation** 但之前hinton的那个工作是本文框架的一种特例, 该方法是有train/test data和white box的teacher(victim)模型作为先验知识的
- 本文就黑盒模型偷取提出4个问题 :

  1） 能不能使用随机的请求图像和其相应的预测训练knockoff

  2） 什么是好的请求数据集

  3） 怎样提升样本请求的效率

  4） 什么是更好的knockoff结构

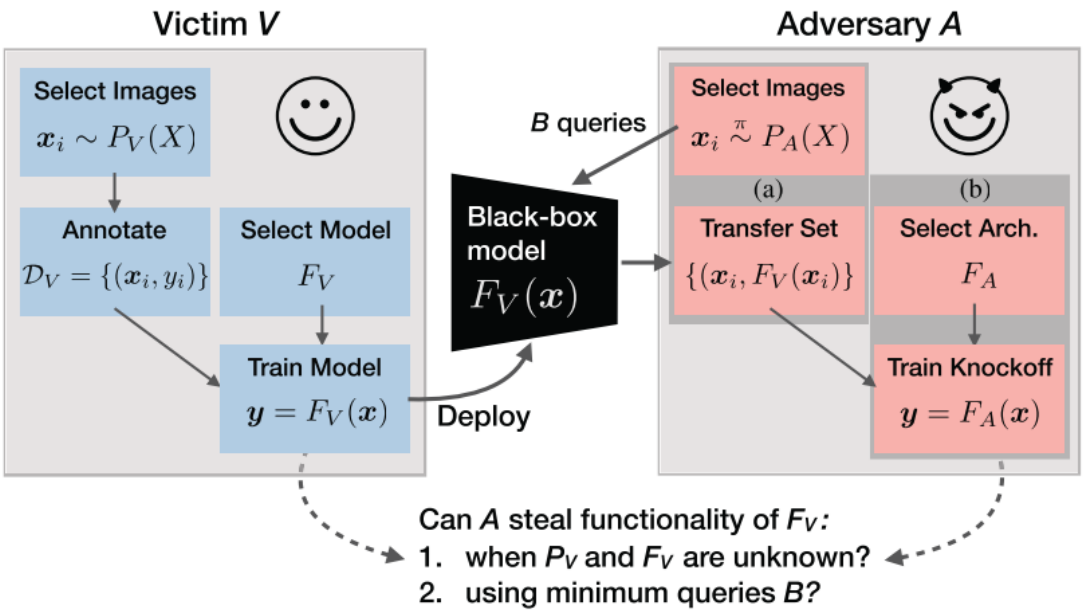


Figure 2: Problem Statement. Laying out the task of model functionality stealing in the view of two players - victim V and adversary A. We group adversary's moves into (a) Transfer Set Construction (b) Training Knockoff F_A.

Method

Functionality Stealing : 给出可以进行请求的黑盒victim model F_v : $X->Y$, 可以使用一个knockoff仿制品模型复制其相应的功能F_v

Victim's Move : victim model的目标是使用训练集训练好一个模型, 然后部署到实际场景中进行使用.

Adversary's Unknowns : 攻击者只知道victim model是一个classifier, 给定任何图片会返回一个K-dim的后验概率. 同时有关victim model的内部信息, 如结构和超参数等不知道; 训练集和测试集不知道; K个类别的语义不知道

Adversary's Attack : 为了训练仿造品模型, 攻击者会 : (1)交互的对victim model使用从strategy π生成的数据得到一个transfer set进行请求数据的pseudo-label. (2)选择一个F_A的结构作为仿制品的结构, 然后使用这个数据集训练F_A, 并在transfer set中模仿F_V的行为

Objective : 本文主要关注与攻击者, 其主要任务是训练一个能够在F_V的任务上同样表现良好的替代模型. 额外的, 还有两个次要目标: (1)采样效率问题, 在有限的请求里最大化模型的性能. (2)了解什么是请求黑盒模型最好的数据

和Knowledge Distillation的区别 : 不同的采样数据分布 P_A , 与训练victim model的数据分布不同. KD是supervision的方式, student模型最小化cross-entropy在原始数据集上和新数据集上, 并且2项loss都包含了参数.

$$\mathcal{L}_{\mathrm{KD}} = \lambda_1 \mathcal{L}_{\mathrm{CE}}(\boldsymbol{y}_{\mathrm{true}}, \boldsymbol{y}_S) + \lambda_2 \mathcal{L}_{\mathrm{CE}}(\boldsymbol{y}_S^\tau, \boldsymbol{y}_T^\tau) \qquad (1)$$

where $\boldsymbol{y}_T^\tau = \mathrm{softmax}(\boldsymbol{a}_T/\tau)$ is the softened posterior distri-

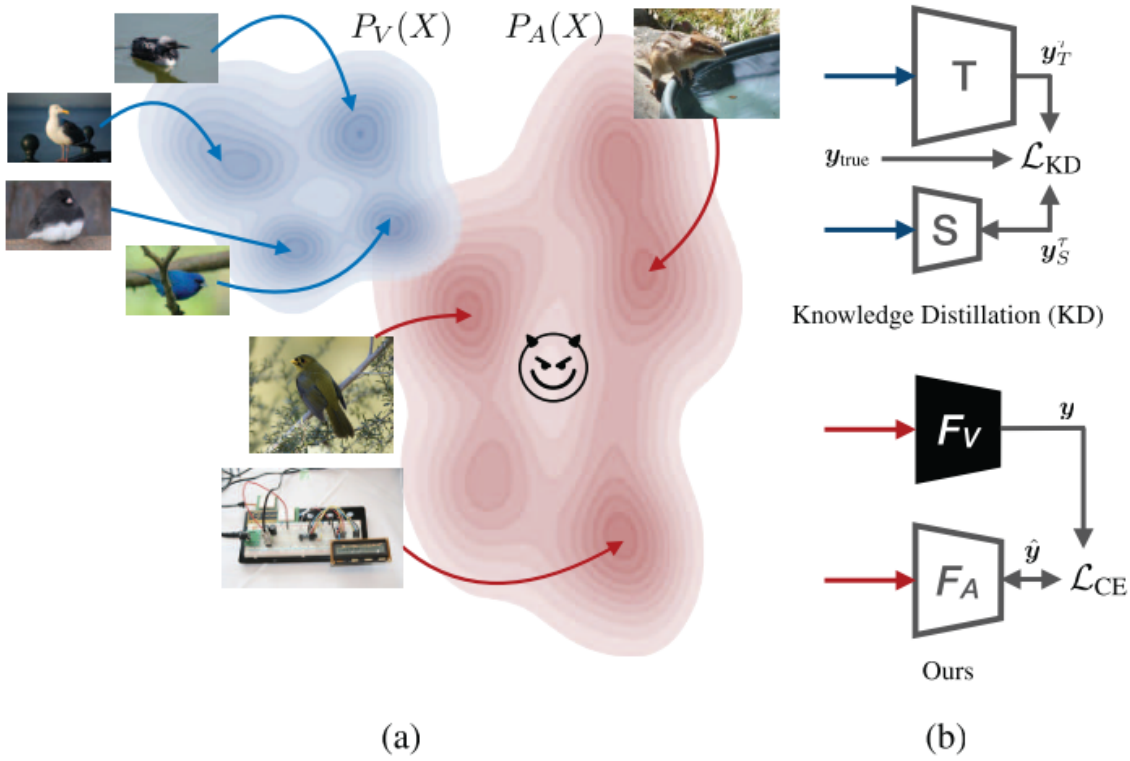


Figure 3: Comparison to KD. (a) Adversary has access only to image distribution $P_A(X)$ (b) Training in a KD-manner requires stronger knowledge of the victim. Both S and F_A are trained to classify images $\boldsymbol{x} \in P_V(X)$

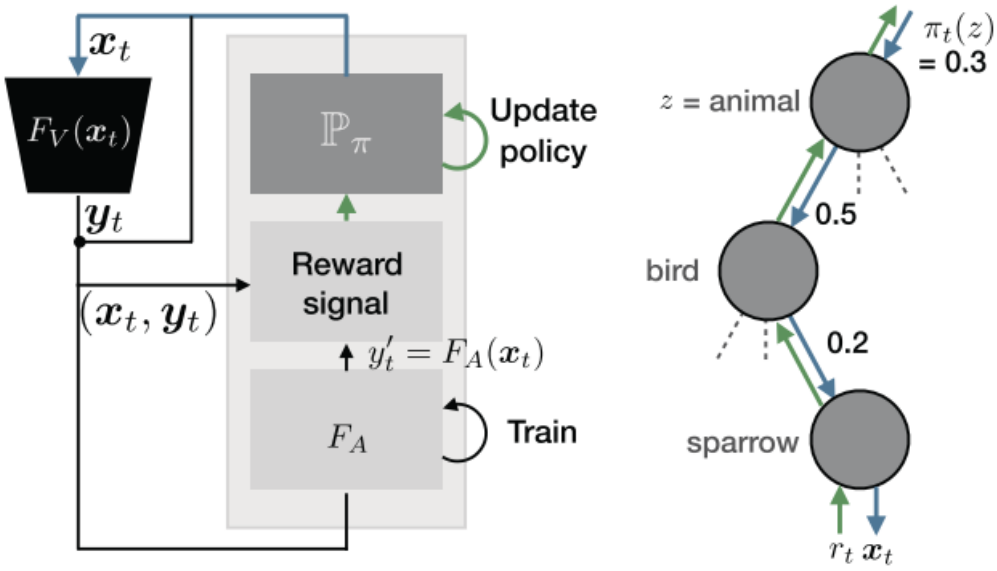
Generating Knockoffs

本文方法生成knockoff分为两个步骤 : Transfer Set Construction和训练knockoff

Transfer Set Construction

- 选择一个合适的$P_A(X)$: 考虑可以用大的公开数据集当做采样分布, 比随机分布好得多
- 采样策略π : 考虑使用两种策略在$P_A(X)$上进行采样. 1) Random Strategy, 有可能采样到和victim model的task完全无关的数据, 导致没有意义. 2) Adaptive Strategy, 使用基于强化学习的方法每次

抽取一小批数据进行对victim model的请求, 对reward最大化来进行更新选择.



(a) Overview (b) Hierarchical policy $\mathbb{P}_\pi$

Figure 4: Strategy `adaptive`.

- 对P_A的补充, 可以通过一些unsupervised方法来获得标签, 从而发现类别之间的相关性.
- 接下来就是使用Adaptive Sample的策略

Training Knockoff F_A

在经过以上的步骤后, 生成了一个transfer set, 然后需要考虑两点: 1 knockoff的结构, 2 训练细节

- knockoff的结构, 本文中认为选取一些与相关任务规模相适应的知名网络作为backbone是好的, 甚至可以使用pretrain的backbone
- knockoff的学习. 使用cross-entropy loss

ing the cross-entropy (CE) loss: $\mathcal{L}_{\text{CE}}(\boldsymbol{y}, \hat{\boldsymbol{y}}) = -\sum_k p(y_k) \cdot \log p(\hat{y}_k)$. This is a standard CE loss, albeit weighed with the confidence $p(y_k)$ of the victim's label.

Experiment

Black-Box Victim Model F_V

选择4个不同的fine-grained图像分类CNN网络, 每个网络在一个特定数据集上运行任务. 所有模型都使用ResNet-34结构(使用Image Net Pretrained Weights), 这个结构能够达到很好的性能, 并且是合理的开销. 这些模型被训练好后, 在后续实验中被当做黑盒 : 图片输入, 给出图片分类的类别和概率

P_A的选择

对于random strategy, 不用刻意进行选择, 对于adaptive strategy, 本文中有4种方式 :

$P_A = P_V$ 这个特殊情况和knowledge-distillation一样, 并且temperature=1

$P_A = ILSVRC$ 使用ILSVRC数据集

$P_A = OpenImage$ 使用Open Image数据集

$P_A = D^2$ 使用宇宙中所有数据集, 其实指使用本文实验中所提到的4个数据集, 也就是全部数据

Metrics

$Top-1 Accuracy$: 在victim model的测试集上获得的最大准确率

$SampleEfficiency$: 在预算内最好的性能

Result

		random				adaptive			
	P_A	Caltech256	CUBS200	Indoor67	Diabetic5	Caltech256	CUBS200	Indoor67	Diabetic5
	P_V (F_V)	78.8 (1×)	76.5 (1×)	74.9 (1×)	58.1 (1×)	-	-	-	-
	P_V (KD)	82.6 (1.05×)	70.3 (0.92×)	74.4 (0.99×)	54.3 (0.93×)	-	-	-	-
Closed	D^2	76.6 (0.97×)	68.3 (0.89×)	68.3 (0.91×)	48.9 (0.84×)	82.7 (1.05×)	74.7 (0.98×)	76.3 (1.02×)	48.3 (0.83×)
Open	ILSVRC	75.4 (0.96×)	68.0 (0.89×)	66.5 (0.89×)	47.7 (0.82×)	76.2 (0.97×)	69.7 (0.91×)	69.9 (0.93×)	44.6 (0.77×)
	OpenImg	73.6 (0.93×)	65.6 (0.86×)	69.9 (0.93×)	47.0 (0.81×)	74.2 (0.94×)	70.1 (0.92×)	70.2 (0.94×)	47.7 (0.82×)

Table 2: Accuracy on test sets. Accuracy of blackbox F_V indicated in gray and knockoffs F_A in black. KD = Knowledge Distillation. Closed- and open-world accuracies reported at B=60k.

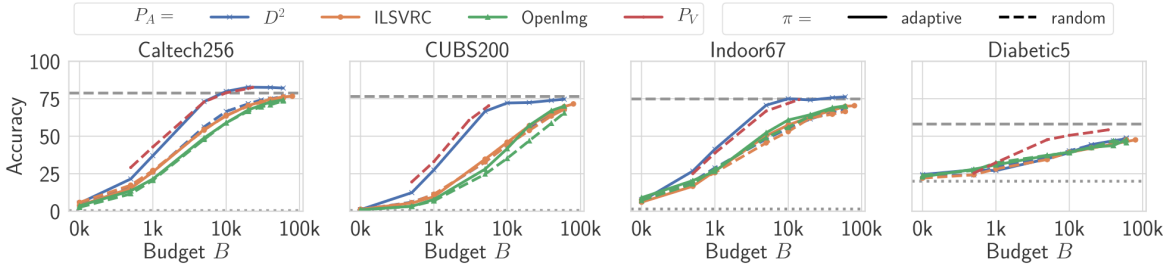


Figure 5: Performance of the knockoff at various budgets. Across choices of adversary's image distribution (P_A) and sampling strategy π. - represents accuracy of blackbox F_V and ····· represents chance-level performance. Enlarged version available in supplementary.

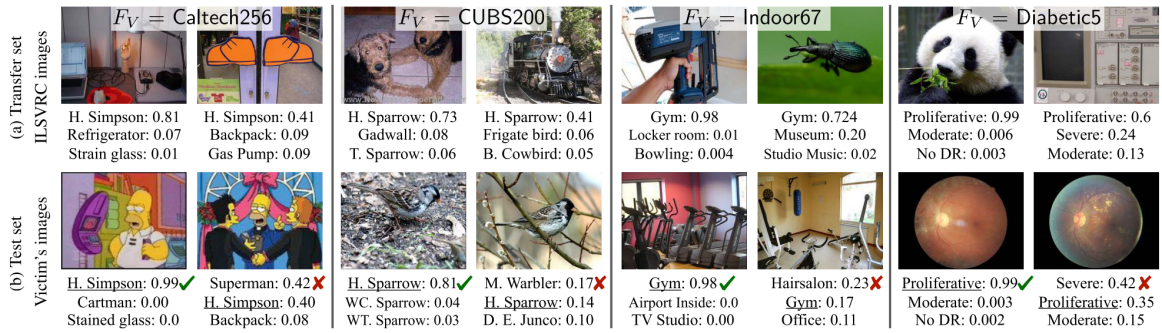


Figure 6: Qualitative Results. (a) Samples from the transfer set ($\{(\boldsymbol{x}_i, F_V(\boldsymbol{x}_i))\}, \boldsymbol{x}_i \sim P_A(X)$) displayed for four output classes (one from each blackbox): 'Homer Simpson', 'Harris Sparrow', 'Gym', and 'Proliferative DR'. (b) With the knockoff F_A trained on the transfer set, we visualize its predictions on victim's test set ($\{(\boldsymbol{x}_i, F_A(\boldsymbol{x}_i))\}, \boldsymbol{x}_i \sim \mathcal{D}_V^{\text{test}}$). Ground truth labels are underlined. Objects from these classes, among numerous others, were never encountered while training F_A.

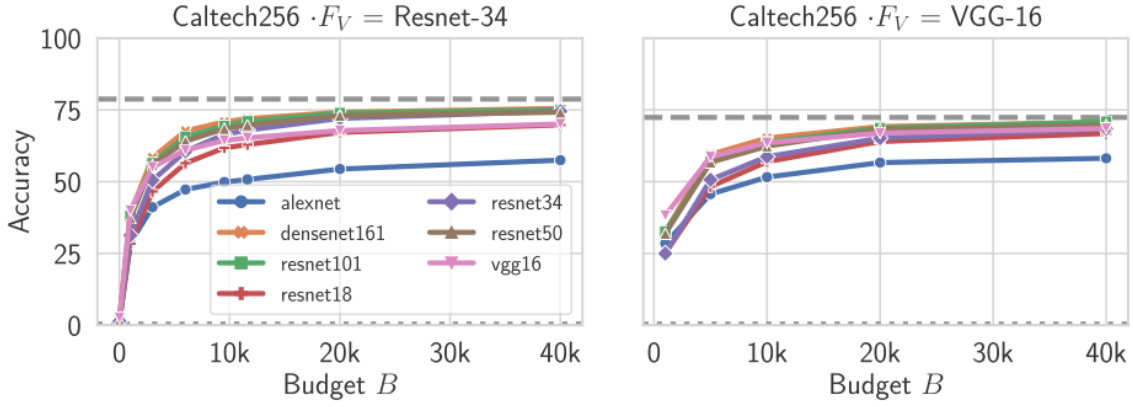


Figure 10: Architecture choices. F_V (left: Resnet-34 and right: VGG-16) and F_A (lines in each plot).

Conclusion

再一次验证了模型抽取的有效性. 对于软工来说, 是完全可能通过模型抽取, 然后做white box transfer, 再替换原有模型的. 本文中就是使用cross entropy进行模型抽取, 然后提出了采样分布选择问题, 并且使用RL的方法去做最有效的采样分布. 并且本文中还提到了采样数据分布的和victim model数据分布的相似度问题, 因为即使两个数据集毫无关系, 模型依然会做出分类结果. 因此要获得一个好的抽取模型, 应当选择相似与原有模型的数据分布来采样数据. 对于我的工作来说, 在operational domain采样的数据集大多是相似但不IID的, 因此这个问题能很好的被解决