GAMIN : An Adversarial Approach to Black-Box Model Inversion

Generative Adversarial Model INversion

Overview

- 总体来说, GAMIN由两个神经网络组成. 一个generator $G:Z->X$ 学习将noise z~$N(0,1)$ 映射到一个x_G. 一个surrogate model $S:X->Y$ 输出一个对target model输出的一个估计$\hat{y}$
- 在对模型S之上进行MIA的时候可以同时训练S和G. 因此generator的目标是学习到与类别相关y_t对应输入x_t的分布
- 算法流程 :

Algorithm 1 GAMIN training

Require: $(G,\ \theta_G)$ generator model, $(S,\ \theta_S)$ shadow model, T target model, y_t target label to invert, k_0 initial boundary-equilibrium factor, λ_k, γ_k boundary-equilibrium update parameters

$k \leftarrow k_0$

for n epochs **do**

▷ Generate artificial inputs from noise

$Z_G \sim \mathcal{N}(0,1)$

$X_G \leftarrow G(Z_G)$

▷ Generate raw noise input

$X_S \sim \mathcal{N}(0,1)$

▷ Query from the target model

$Y_G \leftarrow T(X_G)$

$Y_S \leftarrow T(X_S)$

▷ Compute boundary-equilibrium loss and train surrogate

$L_S \leftarrow L_H(X_S, Y_S) - k * L_H(X_G, Y_G)$

$\theta_S \leftarrow train(X_S, Y_S, L_S)$

$k \leftarrow k + \lambda_k(\gamma_k L_H(X_S, Y_S) - L_H(X_G, Y_G))$

▷ Train generator

$Z_G \sim \mathcal{N}(0,1)$

$\theta_{S \circ G} \leftarrow train(Z_G, y_t)$

end for

对于算法的每一步, 从随机噪声$N(0,1)$中采样一个batch的Z_G. 然后generator使用这个Z_G产生对应一个batch的X_G. 随后分别使用X_G和X_S分别对target model做query, 得到对应的输出预测. 随后组合模型输出的预测构建X_G和X_Z的数据集, 计算surrogate loss对S进行训练. 最后通过combined model$S-G$进行对generator的更新. **在收敛后, surrogate model学习到target model的决策边界, generator学习到对原始数据x_t的近似**.

- 考虑到black-box agnostic attack的设定, GAMIN所选择网络的结构必须是非常普通范用的, 准确的说, GAMIN唯一的限制就是generator的输出要和target model的输入一致

Details

GAMIN的训练过程需要交替的训练surrogate model和generator, 每个网络都有它自己的loss. 但是一个部分的训练会影响到另一个网络的性能, 为了解决这个问题, 需要设计合适的loss函数来更好的控制训练过程

- Boundary-equilibrium loss for surrogate model

  使用BEGAN的Boundary-Equilibrium Adaptive loss. 主要的想法是使用这个loss函数实现自我更新去反应从generated数据和原本noise数据之间的trade off.

  surrogate loss被定义为:

$$L_S = L_H(X_S, Y_S) - k_t * L_H(X_G, Y_G)$$
$$k_{t+1} = k_t + \lambda_k(\gamma_k L_H(X_S, Y_S) - L_H(X_G, Y_G))$$

  其中, L_H是cross-entropy loss

- updating generator model through combined networks

  使用$S(G(z_G))$与真实target model 的cross-entropy计算loss

$$L(y, \hat{y}) = -\sum_i y_i \log(\hat{y}_i)$$

Results

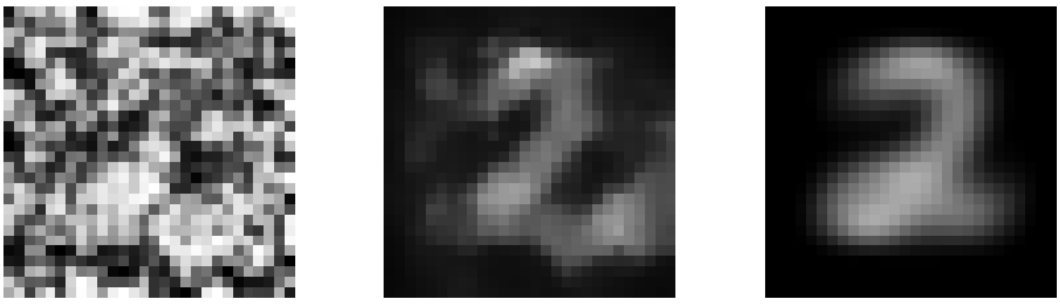

Figure 2: Raw GAMIN output example (left), result of the post-processing on the same attack (center), and mean of the attacked class in original dataset (right).

Conclusion

本文中还是有些看不太明白的东西. 当然也许是作者写的不好. 比如BEGAN LOSS. 本文提出的模式很有意思, 并且可以做到black-box agnostic attack的setting. 利用这一点, 在SE中或许也可以提出black-box agnostic transfer learning的setting.

本文中使用一个generator和一个surrogate model, 并且对target model(API)进行多次query得到其输出. 经过同时训练, 可以使得generator通过random noise可以生成近似的原始训练集数据, surrogate model则可以学习到target model 的decision boundary.

对于我的工作来说, 可以通过回溯找到原始数据集中的高置信度数据, 并且也可以通过surrogate model这样的训练方式来学习到target model 的决策边界, 这样即使是black-box的transfer, 一样可以通过模型抽取来获得一个相似的模型, 然后所有的工作都在这个相似的模型上做.