Inversion: a simple defense against self-obfuscation attacks

Siddhartha Datta
University of Oxford
siddhartha.datta@cs.ox.ac.uk

Nigel Shadbolt
University of Oxford
nigel.shadbolt@cs.ox.ac.uk

1 Introduction

We follow-up on the earlier work from Datta & Shadbolt (2022), where it was found that trigger perturbations mapped to pixel generation errors/perturbations can be leveraged by attackers to self-obfuscate themselves in the generated image output of generative models. Real-world surveillance systems may wish to minimize hardware requirements such as colour capture or image resolution, hence post-processing would be a common method to enhance images (e.g. image colourization, resolution enhancement) for subsequent applications such as person/object tracking. In this paper, we summarize a relatively simple defense that can be adopted to resolve this attack setting through the comparison of the generated image against the original image to identify pixel anomalies.

2 Self-obfuscation attack

Though Datta & Shadbolt (2022) empirically evaluated with a backdoor attack where the attacker can poison the training set and thus leverage whitebox information of the model weights, the self-obfuscation attack also retains generality in blackbox settings (similar to an adversarial attack) where the attacker may have an estimation of the training set (e.g. the attacker collects a surrogate dataset of their own) to identify similar trigger perturbations.

The attack executes as follows: ① Given a pre-processing generative model G in the defender's pipeline, there exists a set of inputs $\{x^{'} : y^{'}\}$ where $x^{'} = x + p_{trigger}$ is triggered and $y^{'} = y + p_{obfuscate}$ is obfuscated if x or t contain target class t. The defender trains G on these pairs and learns an association between the distribution of $p_{trigger}$ and the distribution of $p_{obfuscate}$. The attacker may have an approximation of this mapping either from directly backdooring/poisoning the training set, or from collecting their own surrogate dataset. ② During inference, to obfuscate a specific instance t, the attacker introduces perturbations $p_{trigger}$ to render perturbations $p_{obfuscate}$ in the output. Generalized in equation 1, the optimal weight parameters θ of G is constructed by minimizing the loss of triggered $x^{'}$ against obfuscated $y^{'}$.

$$\theta^* := \arg\min_{\theta} \frac{1}{N} \sum_{n=1}^{N} L(G(\theta, x^{'}), y^{'}) \tag{1}$$

To measure the success of self-obfuscation, the attacker measures the divergence between the obfuscated output $y^{'}$ against the clean output y in the regions containing target class t, given the introduction of $p_{trigger}$ in the input. A higher divergence indicates higher degree of self-obfuscation (equation 2).

$$\max ||G(\theta^*, x_{class=t} + p_{trigger}) - G(\theta^*, x_{class=t})|| \tag{2}$$

3 Inversion as a defense

Given the potential proliferation of such an attack, we have identified a relatively simple procedure to mitigate this attack. At this stage we provide a theoretical procedure, and leave validation for future work.

The defense works as follows: ① The defender passes an input $x^{'}$ through their generative model G to return an output $y^{'}$. ② The defender then inverts $y^{'}$ to an approximate input $G^{-1}(y^{'})$. ③ The defender can measure the difference between the inverted input and original input $|G^{-1}(y^{'}) - x^{'}|$ (e.g. if it exceeds a certain threshold, then the input may be re-processed by a different model or require manual inspection, etc). We highlight three examples on how this defense would be implemented.

Resolution enhancement Given a low-resolution image $x^{'}$ and the generated high-resolution image $y^{'}$, the defender can downsample the generated image to the resolution of the original image $G^{-1}(y^{'})$. The inverted image and original image would have substantial anomalies in the self-obfuscated regions.

Low-light enhancement Given a dark image $x^{'}$ and a brightened image $y^{'}$, the defender can reduce the brightness of the generated image at the measured level of the original image $G^{-1}(y^{'})$ and compute the difference.

Colour enhancement Given a black-and-white image $x^{'}$ and a colourized image $y^{'}$, the defender can reduce gray-scale the generated image $G^{-1}(y^{'})$ and compute the difference.

4 Conclusion

Though simple, inverting the generated output is a straightforward and practical strategy to mitigate the risk of self-obfuscation attacks or other potential tampering attacks against the outputs of generative models. We also encourage future attention on the safety of machine learning models embedded in systems.

References

Siddhartha Datta and Nigel Shadbolt. Hiding behind backdoors: Self-obfuscation against generative models, 2022. URL https://arxiv.org/abs/2201.09774.