Reading Notes on defenses

1 TSS: Transformation-Specific Smoothing for Robustness Certification

Semantic Transformation Attacks (e.g. Rotate, Contrast, Brightness, etc.)

- Often not constrainted by norm metrics
- Forms a very low dimension space (comparing with the image space)
- Preserve semantic information
- Some may introduce interpolation error

1.1 Formalizing transformation:

- Consider the image set $X \subseteq \mathbb{R}^d$ and label set $Y \subseteq [1..c]$.
- Let $\mathcal{Z} \subseteq \mathbb{R}^m$ be the space of transformation parameters.
- Define a transformation as a mapping $\phi : X \times \mathcal{Z} \to X$.

For an arbitrary classifier $h(x)$, we denote its ϵ-smoothed classifier as:

$$g(x;\epsilon) = \arg\max_{y \in Y} \mathbb{E}_{\epsilon \sim \mathbb{P}_\epsilon}(p(y|\phi(x,\epsilon))) \tag{1}$$

1.2 The theorm of robustness for general transformation

- Let $\epsilon_0 \sim \mathbb{P}_0$ and $\epsilon_1 \sim \mathbb{P}_1$ be random variables in $\mathcal{Z}$, with pdf f_0, f_1.
- Let $y_A = g(x;\epsilon_0)$ where g is a smoothed classifier of $\phi : X \times \mathcal{Z} \to X$.
- Suppose there is a probability gap between the top-2 classes. Specifically, we have
  - $$\mathbb{E}_{\epsilon_0 \sim \mathbb{E}_{\epsilon_0}} p(y_A|\phi(x,\epsilon_0)) \geq p_A > p_B \geq \max_{y \neq y_A} \mathbb{E}_{\epsilon_0 \sim \mathbb{E}_{\epsilon_0}} p(y|\phi(x,\epsilon_0)) \tag{2}$$
- For $t \geq 0$, $\underline{S_t} := \{f_1/f_0 < t\}$ and $\overline{S_t} := \{f_1/f_0 \leq t\}$. Define the function $\xi : [0,1] \to [0,1]$ by:

$$\xi(p) := \sup\{\mathbb{P}_1(S) : \underline{S}_{\tau_p} \subseteq S \subseteq \overline{S}_{\tau_p}\} \tag{3}$$

- where $\tau_p := \inf\{t \geq 0 : \mathbb{P}_0(\overline{S_t}) \geq p\}$.

Then, if $\xi(p_A) + \xi(1-p_B) > 1$, $g(x;\epsilon_0) = g(x;\epsilon_1)$.

Note: Here is a possible intuitive explanation of this theorem.

- Why $\xi(p_A)$ represents the lower bound of $\mathbb{E}_{\epsilon_1} p(y \mid \phi(x,\epsilon_1))$? Consider the problem of minimizing

$$\min \int f_1(\mu) h_A(\mu) \mathrm{d}\mu \quad \text{given} \quad \int f_0(\mu) h_A(\mu) \mathrm{d}\mu = p_A \tag{4}$$

- This is actually a continuous version of the knapsack problem, so a greedy algorithm is optimal!
- We only need to greedily fill $h(\mu) = y_A$ in the ascending order of f_1/f_0. To describe it formally,

$$h(\mu) = y_A \text{ iff } \mu \in S, \text{where } \underline{S}_t \subseteq S \subseteq \overline{S}_t \text{ and } \int_{\mu \in S} f_0(\mu) = p_A. \tag{5}$$

- Therefore, $\min \int f_1(\mu) h_A(\mu) \mathrm{d}\mu = \int_{\mu \in S} f_1(\mu) \mathrm{d}\mu = \mathbb{P}_1(S)$.

1.3 Taxonomy of Semantic Attacks

- Resolvable: if $\forall \alpha \in \mathcal{Z}$, there exists a resolving function $\gamma_\alpha : \mathcal{Z} \to \mathcal{Z}$ that is injective, continuously differentiable with non-vanishing Jacobian, and that

$$\phi(\phi(x, \alpha), \beta) = \phi(x, \gamma_\alpha(\beta)) \quad x \in X, \beta \in \mathcal{Z} \tag{6}$$

  - This equation defines the composition rule of resolvable transformations, so that an attack $\phi(x, \alpha)$ would be smoothed by a ϵ_0 smoother to $\epsilon_1 = \gamma_\alpha(\epsilon_0)$. The general theorem of robustness is thus applicable.
- Differentially Resolvable: if $\forall x \in X$, there exists a resolvable transformation $\psi : X \times Z_\psi \to X$ and a function $\delta_x : Z_\phi \times Z_\phi \to Z_\psi$, such that

$$\phi(x, \alpha) = \psi(\phi(x, \beta), \delta_x(\alpha, \beta)) \tag{7}$$

  - For transformations that are not resolvable (e.g. rotation, which suffers from interpolation error), differentially resolvable is a more general definition for them.
  - It is called *differentially* resolvable since the difference between $\phi(x, \alpha)$ and $\phi(x, \beta)$ is resolvable.
  - Consider $\psi(x, \delta) = x + \delta$. Any ϕ with continuous pixel value changes can be resolved by ψ.

1.4 Theorems of Differentially Resolvable Transformations

- Suppose $\phi : X \times Z_\phi \to X$ is resolved by $\psi : X \times Z_\psi \to X$. Let g be an ϵ-smoothed (in the space of Z_ψ) classifier.
- Consider $S \subseteq Z_\phi$ and define $\{\alpha_i\}_{i=1}^N \subseteq S$ as a set of sampled parameters. Suppose we have found the probability gap that:

$$\mathbb{E}_{\epsilon \sim \mathbb{P}_\epsilon} p(y_A \mid \phi(x, \alpha_i); \epsilon) \geq p_A^{(i)} > p_B^{(i)} \geq \max_{y \neq y_A} \mathbb{E}_{\epsilon \sim \mathbb{P}_\epsilon} p(y \mid \phi(x, \alpha_i); \epsilon) \tag{8}$$

- Then there exists a set $\Delta^* \subseteq Z_\psi$, such that if $\forall \alpha \in S, \exists \alpha_i$ with $\delta_x(\alpha, \alpha_i) \in \Delta^*$, then

$$g(\phi(x, \alpha); \epsilon) = y_A. \tag{9}$$

Note:

- Δ^* is actually the certified space of robustness in $\mathbb{Z}_\psi$, which can be derived by procedures of certifying a resolvable transformation. As stated in *Corollary 2*, when $\psi(x + \delta) = x + \delta$ and ϵ is a Gaussian random variable, it is able to certify a ball area centered at the origin: $\Delta_i \subseteq Z_\psi$. The intersection $\Delta^* = \cap_i \Delta_i$ then naturally satisties the restriction.
- Again consider the case in *Corollary 2*, Δ^* is a ball $B_R(0)$. Then the procedure of sampling $\{\alpha_i\}$ can be viewed as **certifying many small balls** in $\mathbb{Z}_\psi$. Any $\alpha \in S$ can be certified as long as $\phi(x, \alpha)$ is mapped closer enough to a $\phi(x, \alpha_i)$.

1.5 Examples

- Gaussian Blur: take convolution with Gaussian function. $\phi_B(x, \alpha) = x * G_\alpha$. It is not only resolvable, but additive as well.
- Brightness + Contrast. $\phi_{BC}(x, \alpha) = e^k(x + b)$ where $\alpha = (k, b)^T$.

2 Robustness Certification for Point Cloud Models

2.1 About Point Clouds

- Expression of 3D data: 2D projections, 3D voxels (a 3D 0/1 map), meshes, point clouds...
- Point cloud: a sparse representation.
- - Need to be invariant to permutations in the orders of points.
- Note: It seems that in the point cloud scenario, transformations like rotation and scaling are *resolvable* since they do not introduce errors of bilinear interpolation?

2.2 Key idea

1. Use first-order taylor approximation to bound any differentiable transformation function
2. A precise relaxation for global feature poolings, which are more complex than pointwise activation layers such as ReLU.

2.3 Taylor Approximation

- For a point cloud $P = \{p^{(j)} \mid p^{(j)} \in \mathbb{R}^3, j \in \{1, 2, \dots, n\}\}$ and any C^1 transformation $f : \mathbb{R}^{3\times n} \times \mathbb{R}^k \to \mathbb{R}^{3\times n}$, by Taylor expansion at $f(P, t)$:

$$f(P, \theta) = f(P, t) + (\theta - t)^T \cdot \frac{\partial f}{\partial \theta}(P, t) + \frac{1}{2}(\theta - t)^T H(P, \xi)(\theta - t) \tag{10}$$

- for some ξ on the path from t to θ. It is thus only required to bound the second order error $R = \frac{1}{2}(\theta - t)^T H(\theta - t)$.

2.4 Max Pool Relaxation

- Max pooling: $y = \max_i x_i$. Suppose $x_i \in [l_i, u_i]$.
- One can find a trivial lower bound $y \geq \max_i l_i$ and a trivial upper bound $y \geq \max_i u_i$. However, this upper bound causes too much precision loss so a preciser upper bound is required.
- This paper presents an idea of computing the convex hull of all possible cases of $y = x_i$. That is, to compute the convex hull of $\{S_i\}$, where $S_i = \{y = x_i, x_i \geq x_{j\neq i}, x_j \in [l_j, u_j]\}$.

2.5 Specific Transformation

- Rotation: $f(p, \theta)$. A possible representation: rotate around $\hat{\theta}$ for an angle of $\|\theta\|_2$.
  - Rotations are additive when the rotation axis is fixed (and also without interpolation), but not addtive when the axis is arbitrary. Still, it is resolvable when no interpolation is needed.
- Shear: $(x, y, z; \theta) \mapsto (\theta_1 z + x, \theta_2 z + y, z)$.
- Twist, Taper: These are special geometry transformations that occur only in 3D space.
  - These transformations are resolvable only when the main axis is fixed. With random axes, they are not resolvable.
  - To directly apply *Corollary 2* of TSS for differentially resolvable transformations, it seems we need to sample parameters $\{\alpha_i\}$ from $S^2 \times \mathbb{R}$, which may be prohibitively large.

3 End-to-end Robustness for Sensing-Reasoning Machine Learning Pipelines

3.1 Summary

Many previous methods have been proposed to certify the robustness of machine learning models the pertubation bounded in a small ℓ_p ball. In this paper, a generic Sensing-Reasoning machine learning pipelines was proposed, in which the previous methods were viewed as a certification of sensing robustness. The output of sensing (deep learning) models were combined with embedded domain knowledge in reasoning components to provide end-to-end robustness. This pipeline is a generic framework since the choices of specific certified robust sensing models are orthogonal to the certification of reasoning robustness.

The analyses of reasoning robustness started with showing the hardness of certifying the robustness of a general reasoning model. By proving the polynomial time reduction of the counting problem to the robustness problem, certifying the robustness of a general reasoning component was proved to be #P-hard. Despite the hardness of this problem, the author argued the possibility of approximating reasoning robustness for specific graph structures.

Robustness bounds were shown to be certifiable for several reasoning structures including Markov logic networks and Bayesian networks.

3.2 MLN

- To compute the marginal distribution of a given variable v, we need to compute two partition functions. $\Pr(v=1) = Z_1/Z_2$.
- To find a proper bound for the maximization problem for $R(\{p_i(X)+\epsilon_i\}_{i\in[n]})$ could be solved by maximizing $Z_1(\{p_i(X)+\epsilon_i\}_{i\in[n]})$and minimizing $Z_2(\{p_i(X)+\epsilon'\})$. The constraint of $\epsilon=\epsilon'$ was eliminated by adding Lagrangian multipliers i.

3.3 Bayesian Network

- Consider a Bayesian Network with binary tree structure and calculate $P(X=1,\{p_i\})$.

Problem: Why $P(X=1)=\sum_{x_1,x_2}P(1|x_1,x_2)\prod_i p_i^{x_i}(1-p_i)^{1-x_i}$?

- According to the binary tree Bayesian network, we should have joint distribution:

$$P(X,x_1,x_2,\ldots,x_n)=P(X|x_1,x_2)P(x_1|x_3,x,4)P(x_2|x_5,x_6)P(x_3,x_4,\ldots,x_n) \tag{11}$$

- However, the marginal distribution above seems to have assumed independence among $x_1,\ldots,x_n$. Moreover, following the factorization of a binary tree Bayesian network, when given x_1,x_2, X is independent with $x_3,\ldots,x_n$. Thus, with fixed x_1,x_2, it seems that the marginal probability $P(X=1)$ should be independent of $x_3,\ldots,x_n$?
- **In short:** Why should not all random variables of sensing output be leaf nodes?

Paper List

TSS: Transformation-Specific Smoothing for Robustness Certification

Knowledge Enhanced Machine Learning Pipeline against Diverse Adversarial Attacks

End-to-end Robustness for Sensing-Reasoning Machine Learning Pipelines

Certified Robustness to Adversarial Examples with Differential Privacy

MACER: Attack-free and Scalable Robust Training via Maximizing Certified Radius

Provably Robust Deep Learning via Adversarially Trained Smoothed Classifiers

Certified Adversarial Robustness via Randomized Smoothing

On the Effectiveness of Interval Bound Propagation for Training Verifiably Robust Models

MixTrain: Scalable Training of Verifiably Robust Neural Networks

Scaling provable adversarial defenses

TBA