CS 4501/6501 Interpretable Machine Learning

Guest Lecture: Robust Attribution Regularization

Jiefeng Chen
Department of Computer Sciences
University of Wisconsin - Madison
jchen662@wisc.edu

Overview

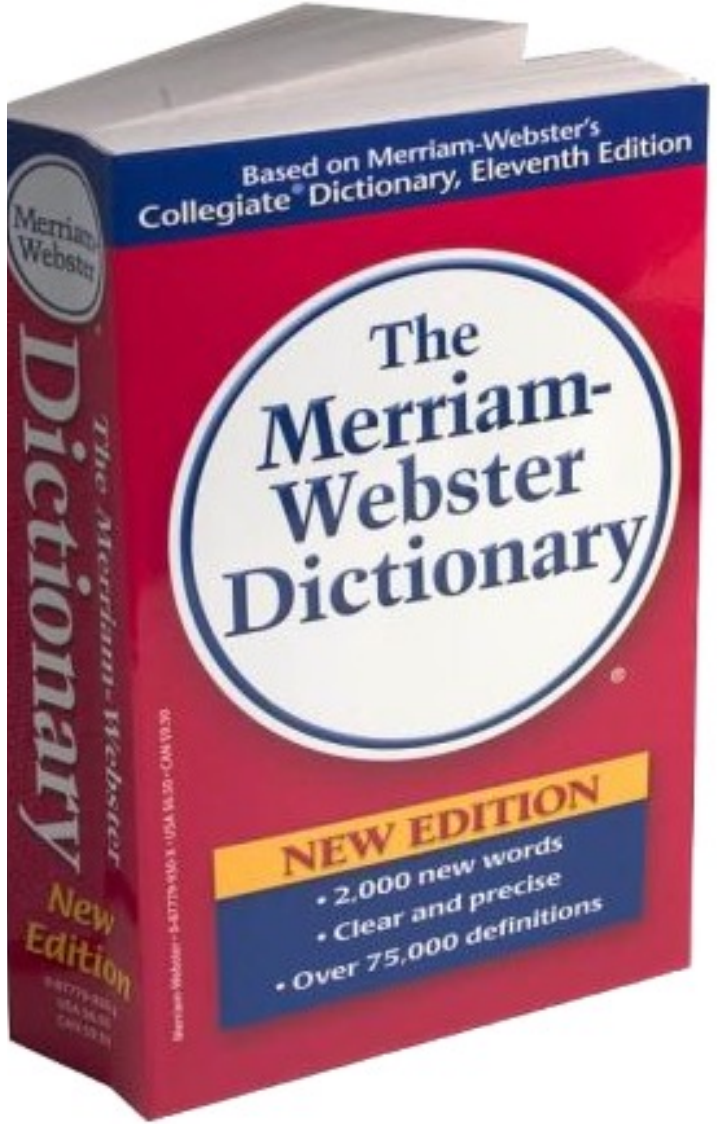
- What is attribution?
- How to compute attribution?
- What is the vulnerability issue of attribution methods?
- Why should attribution methods be robust?
- How to enforce attribution robustness?
- What are the benefits of attribution robustness?

Overview

- What is attribution?
- How to compute attribution?
- What is the vulnerability issue of attribution methods?
- Why should attribution methods be robust?
- How to enforce attribution robustness?
- What are the benefits of attribution robustness?

What is Attribution?

According to Merriam-Webster**:** “*attribution*” means “to explain (something) by indicating a cause”.



Human Learning

Do you see a dog?

Yes.

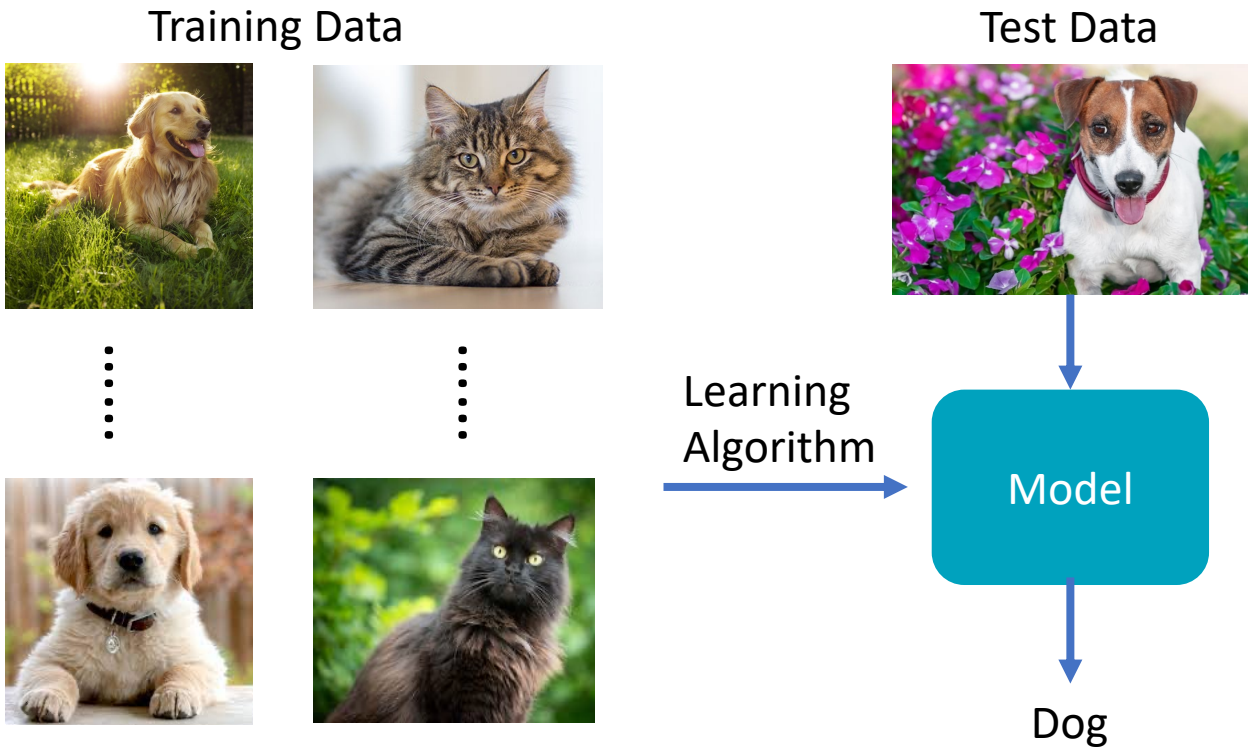
Human Learning

What makes you think so?

Because I see this!

Machine Learning

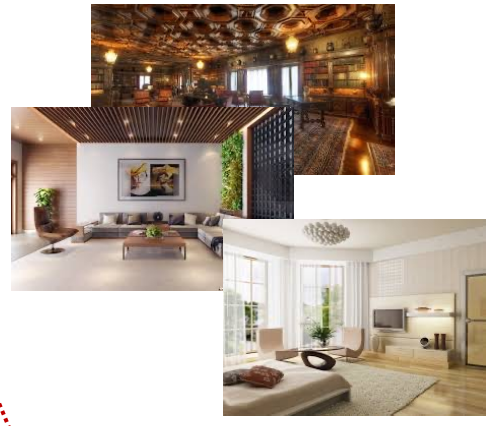
Machine learning is like human learning.



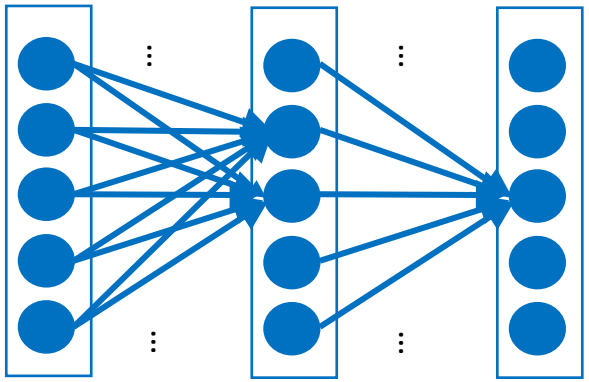
Deep Neural Networks

A neural network with some level of complexity, usually at least two layers, qualifies as a Deep Neural Network (DNN).

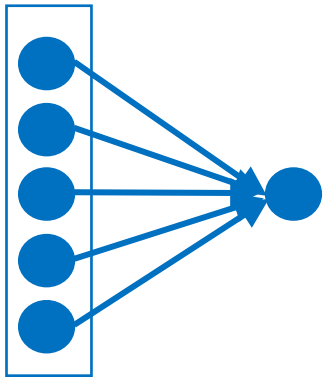
Outdoor



Indoor



... ...



Outdoor

Deep Learning Breakthroughs

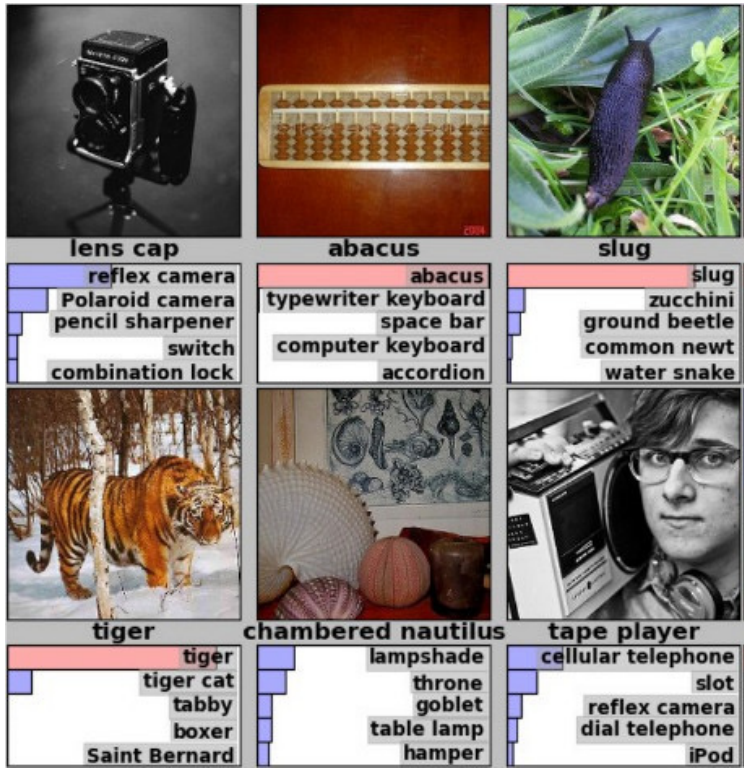


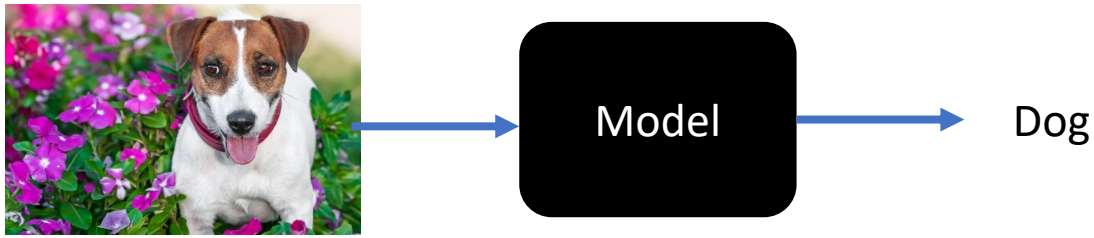
Image Classification

Machine Translation

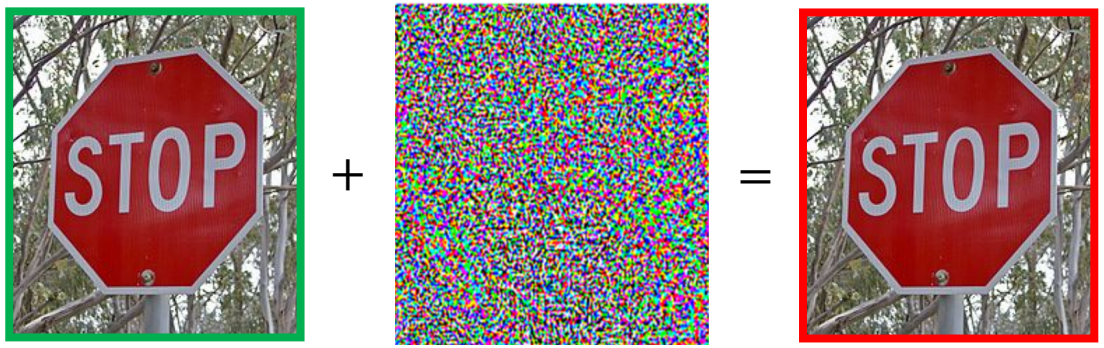
Game Playing

Deep Learning Challenges

- Blackbox: not too much understanding/interpretation



- Vulnerable to adversarial examples



classified as
Stop Sign
97% confidence

classified as
Max Speed 100
98% confidence

Why DNN Models are Vulnerable?

Deep Neural Networks (DNNs) may use *spurious correlation* for prediction.

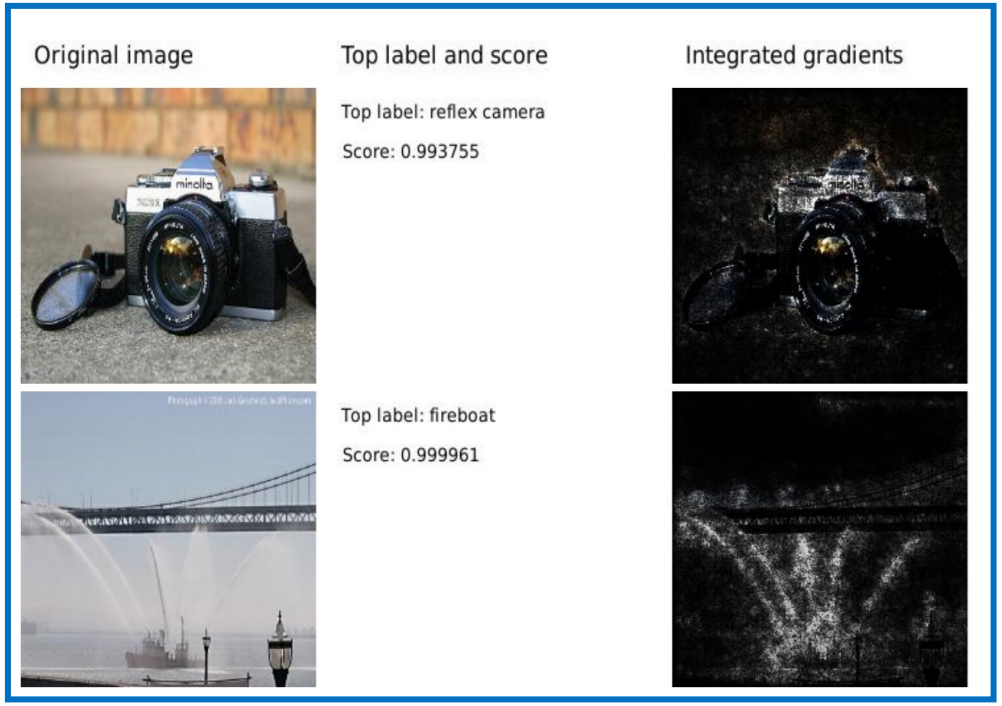
Training Data

How will the DNN behave?

Test Data

Attribution in Machine Learning

Attribution: attributing the prediction of a DNN to its input features.



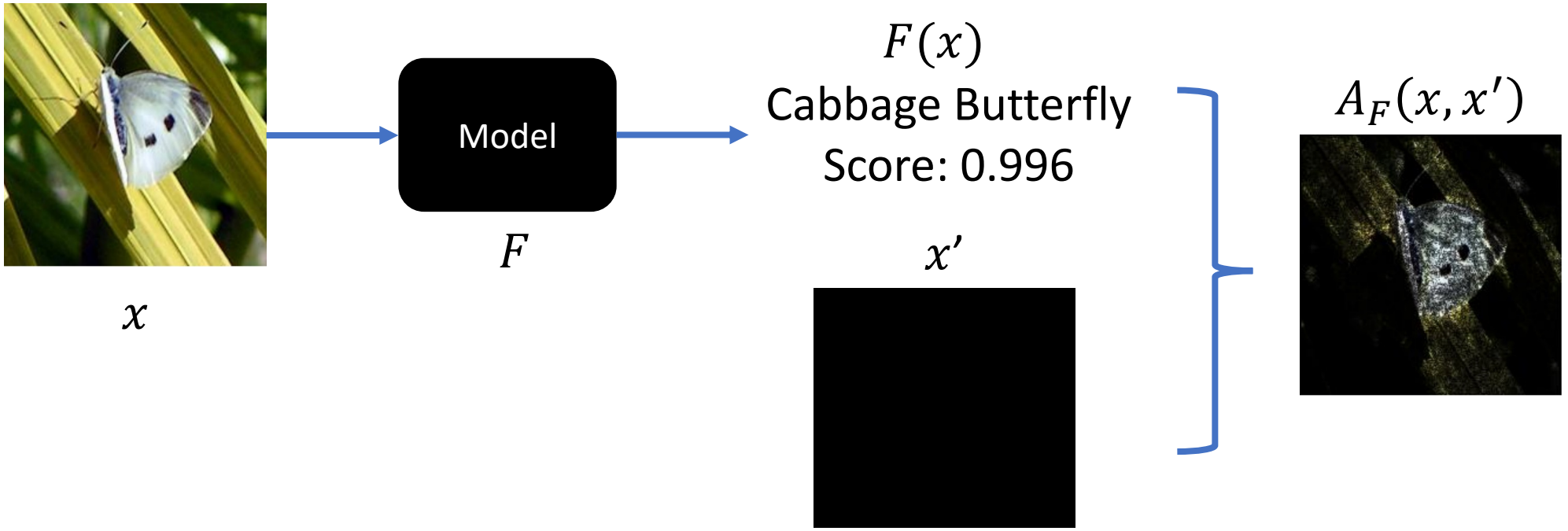
Attributions for Image Classification

how many townships have a population above 50 ? [prediction: NUMERIC]
what is the difference in population between fora and masilo [prediction: NUMERIC]
how many athletes are not ranked ? [prediction: NUMERIC]
what is the total number of points scored ? [prediction: NUMERIC]
which film was before the audacity of democracy ? [prediction: STRING]
which year did she work on the most films ? [prediction: DATETIME]
what year was the last school established ? [prediction: DATETIME]
when did ed sheeran get his first number one of the year ? [prediction: DATETIME]
did charles oakley play more minutes than robert parish ? [prediction: YESNO]

Attributions for Question Classification

Formal Definition of Attribution

Suppose we have a function $F: R^n \rightarrow [0,1]$ that represents a deep network, and an input $x = (x_1, \ldots, x_n) \in R^n$. An attribution of the prediction at input x relative to a *baseline* input x' is a vector $A_F(x, x') = (a_1, \ldots, a_n) \in R^n$ where a_i is the contribution of x_i to the prediction $F(x)$.



A note on the baseline

- The need for a baseline is central to any explanation method. In a sense, it is the counterfactual for causal reasoning.
- The network must have a truly neutral prediction at the baseline input.

Question?

Overview

- What is attribution?
- How to compute attribution?
- What is the vulnerability issue of attribution methods?
- Why should attribution methods be robust?
- How to enforce attribution robustness?
- What are the benefits of attribution robustness?

Common Attribution Methods

- Gradients
- DeepLift
- Layer-wise Relevance Propagation (LRP)
- Deconvolutional Networks
- Guided Back-propagation
- Integrated Gradients

How to evaluate an attribution method?

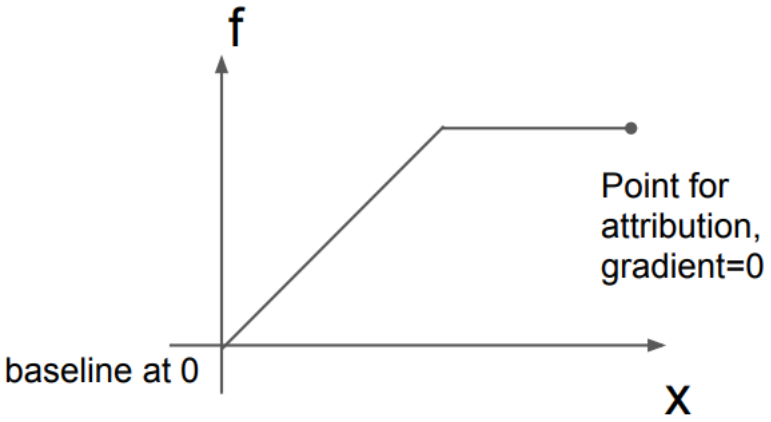
Sundararajan et al.'s Approach:

- Define a set of reasonable axioms for attribution methods.
- Check if the attribution method satisfies them.

Sundararajan, Mukund, Ankur Taly, and Qiqi Yan. "Axiomatic attribution for deep networks." ICML, 2017.

Sensitivity Axioms

- **Sensitivity**: If starting from baseline, varying a variable changes the output, then the variable should receive some attribution.
- **Insensitivity**: A variable that has no effect on the output gets no attribution.



Pure gradients do not satisfy Sensitivity when predictions saturate.

Functional Axioms

- **Implementation Invariance**: Two functionally equivalent networks have identical attributions for all inputs and baseline.
- **Linearity**: If the function F is a linear combination of two functions F_1, F_2, then the attributions for F are a linear combination of the attributions for F_1, F_2.
- **Symmetry**: If a function is symmetric across two input variables then the variables should receive identical attribution.

An Accounting Axiom

Completeness: Sum(attributions) = F(input) - F(baseline).

Break down the predicted click through rate (pCTR) of an ad like:

- 55% of pCTR is because it's at position 1.
- 25% is due to its domain (a popular one).

…

Integrated Gradients (IG)

- The integrated gradient (IG) along the i^{th} dimension for an input x and baseline x' is defined as follows:

$$IG_i(x) \coloneqq (x_i - x_i') \times \int_{\alpha=0}^{1} \frac{\partial F(x' + \alpha \times (x - x'))}{\partial x_i} d\alpha$$

- IG satisfies the *completeness* axiom: if F is differentiable almost everywhere, then $\sum_{i=1}^{n} IG_i(x) = F(x) - F(x')$.

Sundararajan, Mukund, Ankur Taly, and Qiqi Yan. "Axiomatic attribution for deep networks." ICML, 2017.

Theoretical Result

Theorem: Integrated Gradients is the unique method satisfying:

- Sensitivity, Insensitivity
- Implementation Invariance, Linearity, Symmetry
- Completeness

up to the errors from approximating integration.

Implementation of IG

- The integral of IG can be efficiently approximated via a summation:

$$IG_i^{approx}(x) := (x_i - x_i') \times \sum_{k=1}^{m} \frac{\partial F\left(x' + \frac{k}{m} \times (x - x')\right)}{\partial x_i} \times \frac{1}{m}$$

Here, m is the number of steps in the Riemman approximation of the integral.

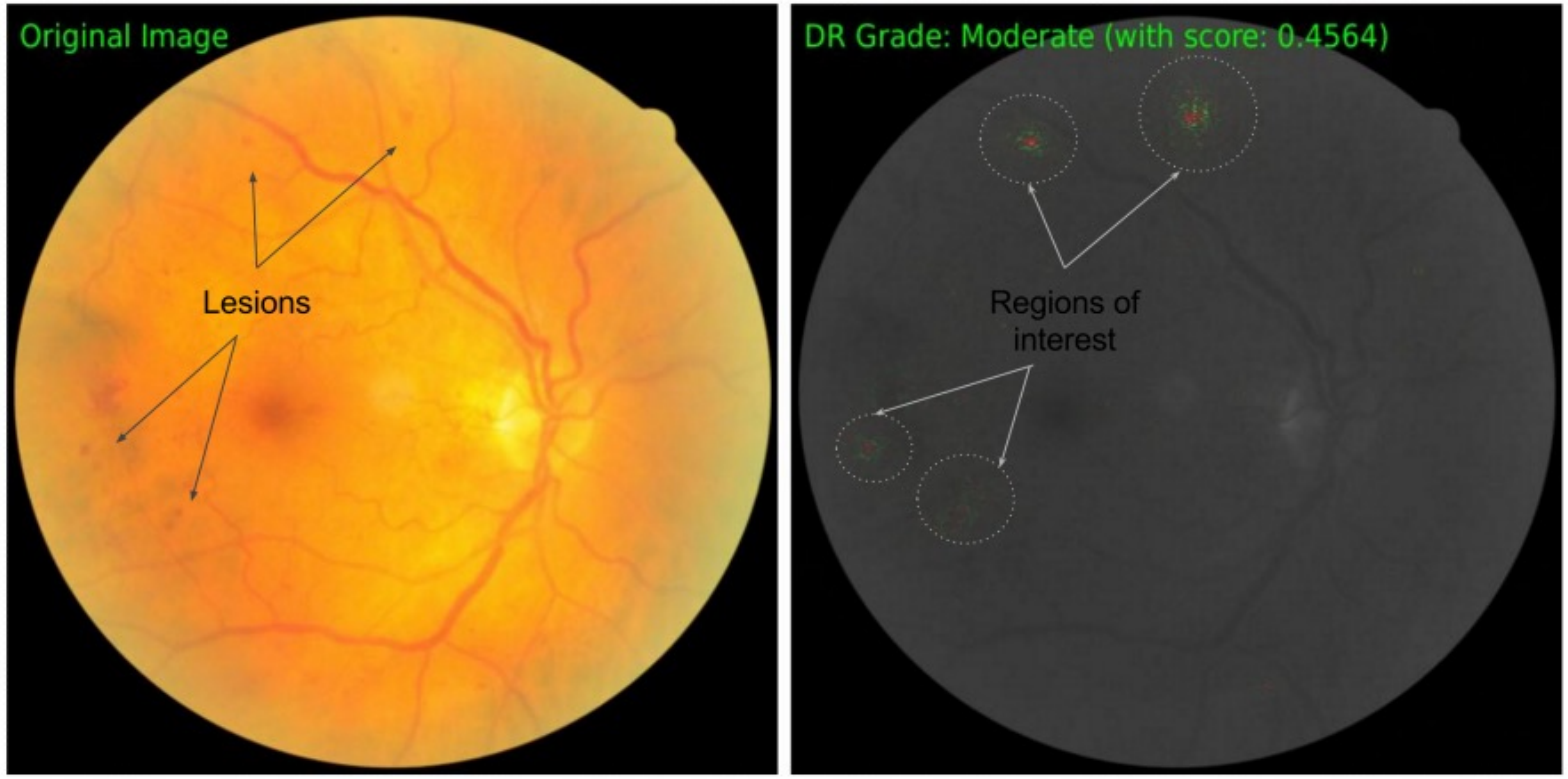
- Step-size m: check if *completeness* holds. If not, increase m.
- Baseline x': select x' that leads to a near-zero score.

TensorFlow Implementation of IG

```
def integrated_gradients(inp, baseline, label, steps=range(50)):
    t_input = input_tensor()  # input tensor
    t_prediction = prediction_tensor(label) # output tensor
    t_gradients = tf.gradients(t_prediction, t_input)[0] # gradients
    path_inputs = [baseline + (i/steps)*(inp-baseline) for i in steps]
    grads = run_network(t_gradients, path_inputs)
    return (inp-baseline)*np.average(grads, axis=0)  # integration
```

Applications of IG

We can use IG for Diabetic Retinopathy Prediction where feature importance explanations are important for specialists to build trust in the network's predictions.



Attribution for Diabetic Retinopathy grade prediction from a retinal fundus image.

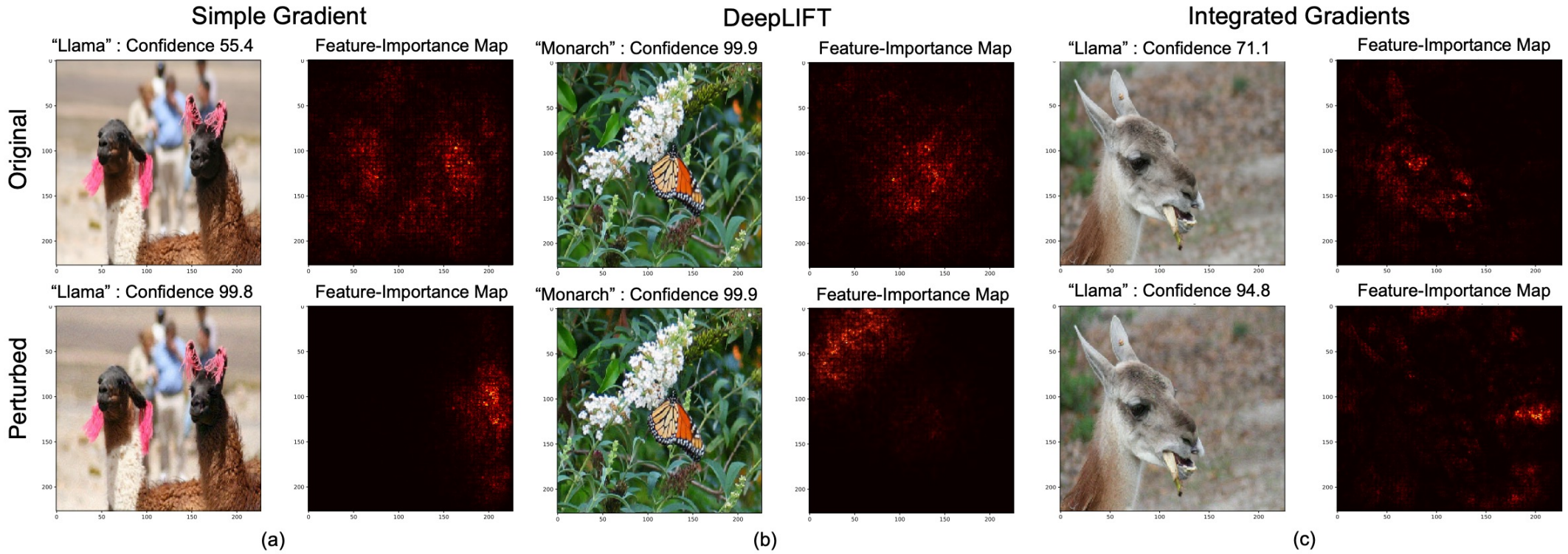
Question?

Overview

- What is attribution?
- How to compute attribution?
- What is the vulnerability issue of attribution methods?
- Why should attribution methods be robust?
- How to enforce attribution robustness?
- What are the benefits of attribution robustness?

Fragile Interpretation

As Ghorbani et al. convincingly demonstrated, for existing DNNs, one can generate minimal input perturbations that substantially change model attributions, while keeping their (correct) predictions intact.



Ghorbani, Amirata, Abubakar Abid, and James Zou. "Interpretation of neural networks is fragile." AAAI 2019.

Attribution Attack Objective

For a given neural network $\mathcal{N}$ with fixed weights and a test data point x_t, the feature importance method produce an interpretation $I(x_t; \mathcal{N})$, which is a vector of normalized feature scores. The attribution attack objective is:

$$\arg\max_{\delta} \mathcal{D}(I(x_t; \mathcal{N}), I(x_t + \delta; \mathcal{N}))$$

$$subject\ to: \left|\left|\delta\right|\right|_{\infty} \leq \epsilon$$

$$Prediction(x_t + \delta; \mathcal{N}) = Prediction(x_t; \mathcal{N})$$

where $\mathcal{D}(\cdot)$ measures the change in interpretation and $\epsilon > 0$ constrains the norm of the perturbation.

Solving Attribution Attack Objective

Algorithm Iterative Feature Importance Attacks

Input: test image x_t, perturbation budget ϵ, normalized feature importance function I(·), number of iterations P, step size α.

Define a dissimilarity function D to measure the change between interpretations of two images:

$$D(x_t, x) = \begin{cases} -\sum_{i \in B} I(x)_i & \text{for } \textbf{top-k} \text{ attack} \\ \sum_{i \in A} I(x)_i & \text{for } \textbf{targeted} \text{ attack} \\ \left|\left|C(x) - C(x_t)\right|\right|_2 & \text{for } \textbf{mass-center} \text{ attack} \end{cases}$$

where B is the set of the k largest dimensions of $I(x_t)$, A is the target region of the input image in targeted attack, and $C(\cdot)$ is the center of feature importance mass.

Initialize $x^0 = x_t$

for $p \in \{1, \dots, P\}$ **do**

Perturb the test image: $x^p = x^{p-1} + \alpha \cdot sign(\nabla_x D(x_t, x^{p-1}))$

If needed, clip the perturbed input to satisfy: $\left|\left|x^p - x_t\right|\right|_\infty \leq \epsilon$

end for

Among $\{x^1, \dots, x^P\}$, return the element with the largest value for the dissimilarity function and the same prediction as the original test image.

Metrics for interpretation similarity

- **Spearman's correlation**: use the rank correlation to compare the similarity between two attributions.
- **Top-k intersection**: compute the size of intersection of the k most important features of the two attributions divided by k.

$$\rho = 1 - \frac{6 \sum d_i^2}{n(n^2 - 1)}$$

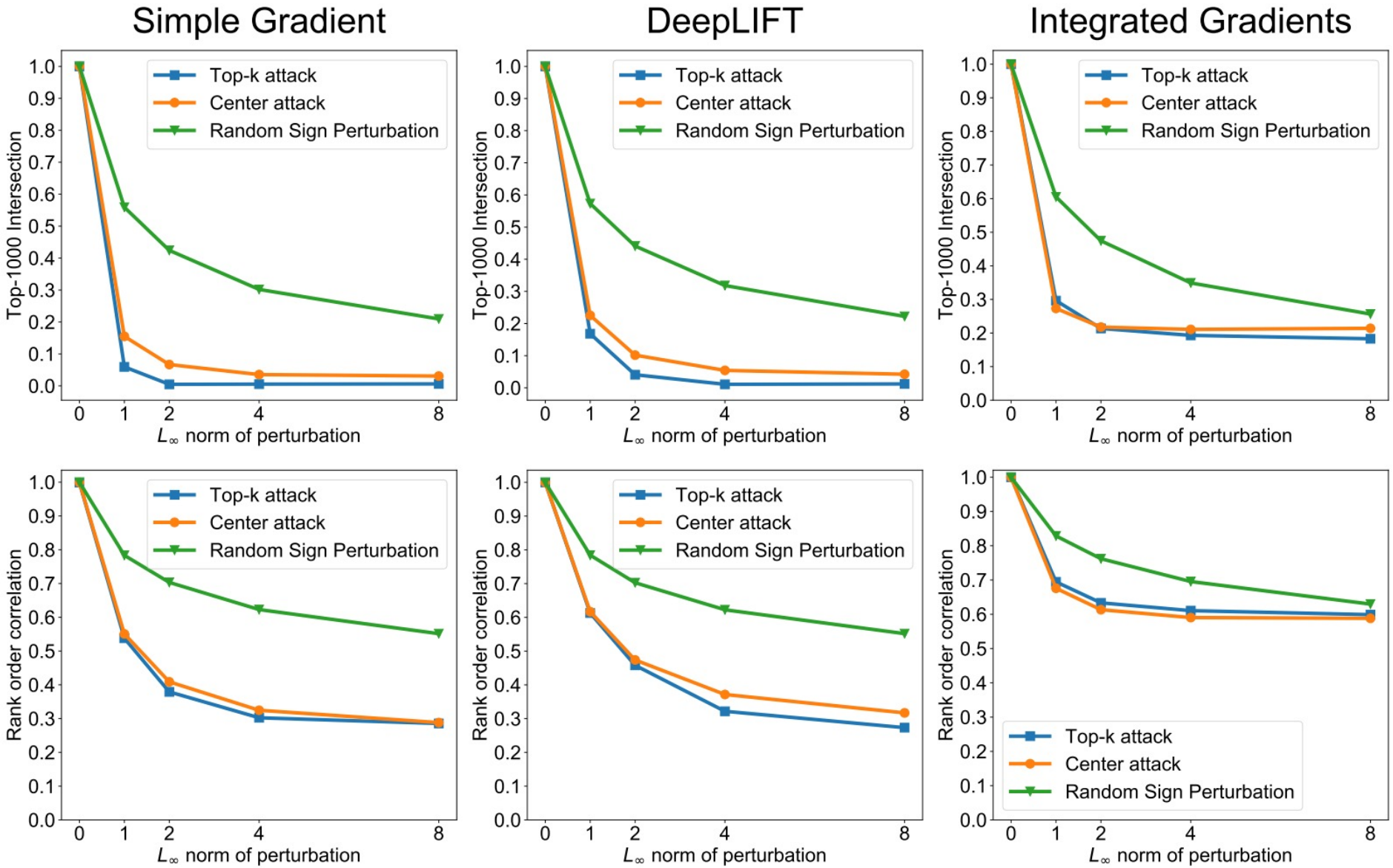
ρ = Spearman's rank correlation coefficient

d_i = difference between the two ranks of each observation

n = number of observations

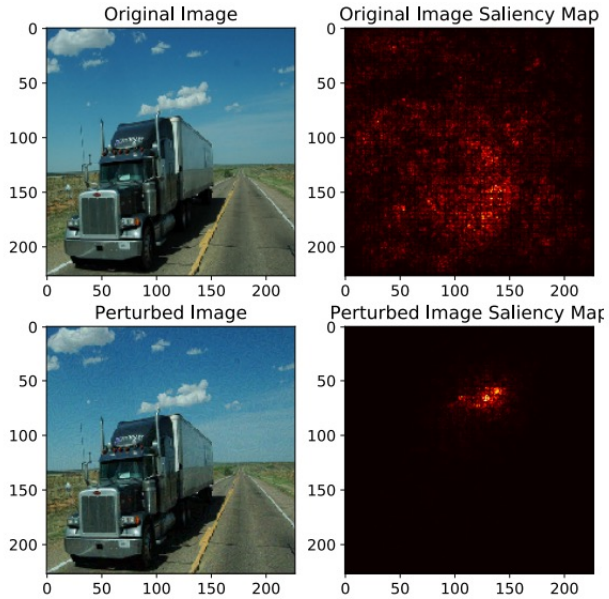
The indices of the 5 most important features of attribution A is {0, 1, 2, 3, 4} while the indices of the 5 most important features of attribution B is {3, 4, 5, 6, 7}. Then the **top-5 intersection** of A and B is 0.4.

Top-k and Mass-center Attack Results

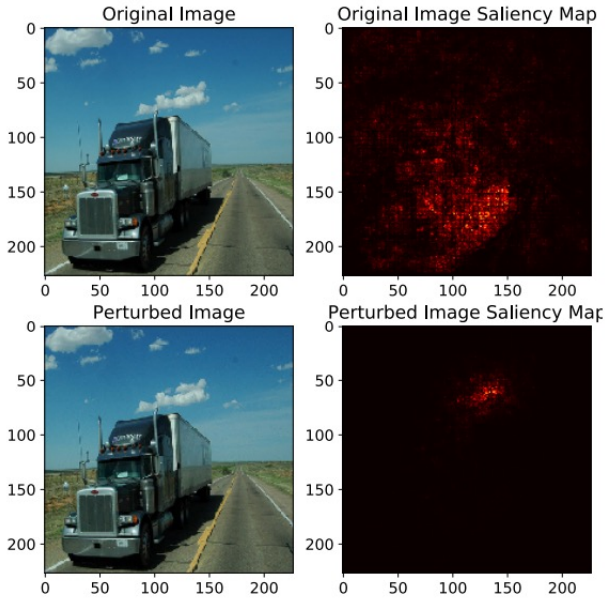


Targeted Attack Results

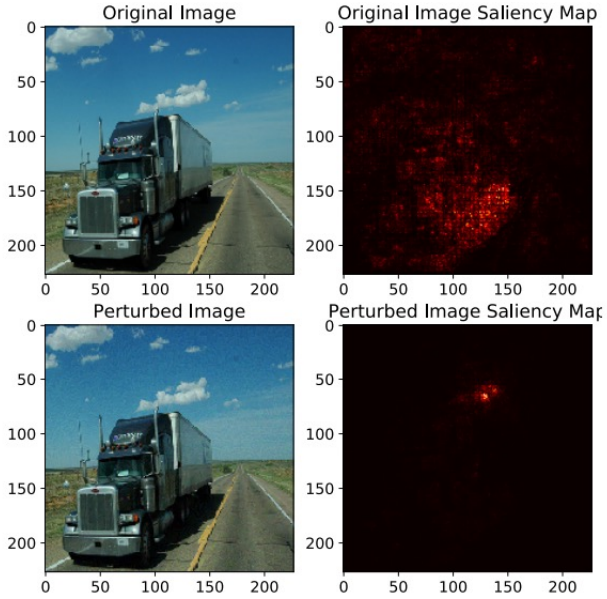
Simple Gradient



DeepLIFT



Integrated Gradients



Question?

Overview

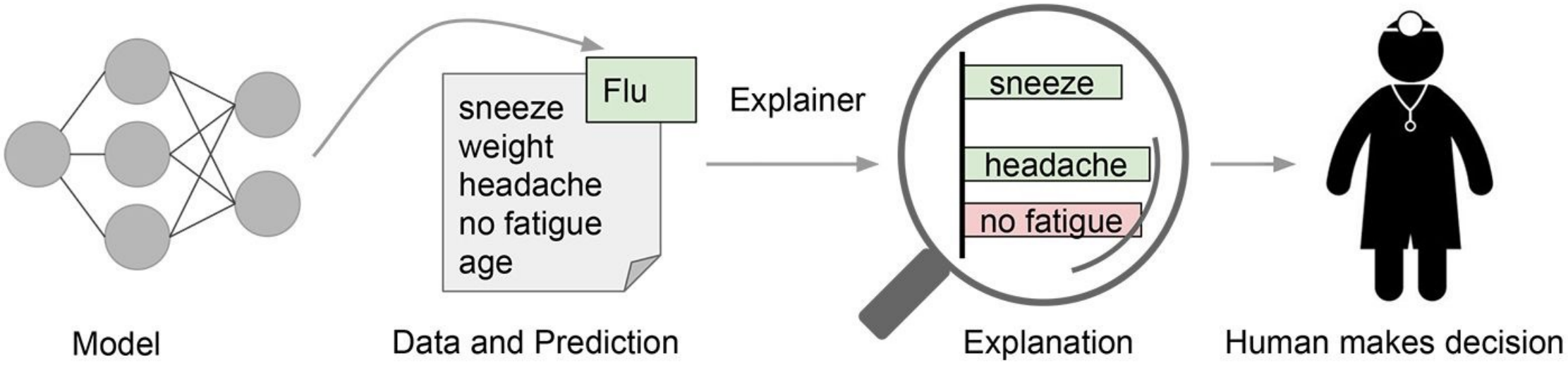
- What is attribution?
- How to compute attribution?
- What is the vulnerability issue of attribution methods?
- Why should attribution methods be robust?
- How to enforce attribution robustness?
- What are the benefits of attribution robustness?

Why we want robust attributions?

Model attributions are *facts* about model behaviors. While robust attribution does not necessarily mean that the attribution is correct, a model with *brittle attribution* can never be trusted.

Importance of Attribution Robustness

In safety critical applications, the users need to check the attribution to see whether the model's predictions could be trusted. If the attributions are brittle, the users will find it difficult to trust the model.



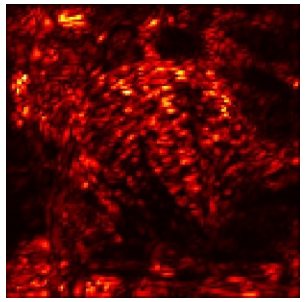
Overview

- What is attribution?
- How to compute attribution?
- What is the vulnerability issue of attribution methods?
- Why should attribution methods be robust?
- How to enforce attribution robustness?
- What are the benefits of attribution robustness?

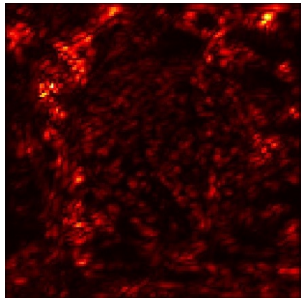
Robust Prediction Correlates with Robust Attribution: Why?

Empirical results demonstrate that if the model has robust prediction, usually it also has robust attribution.

original image,
normally trained model



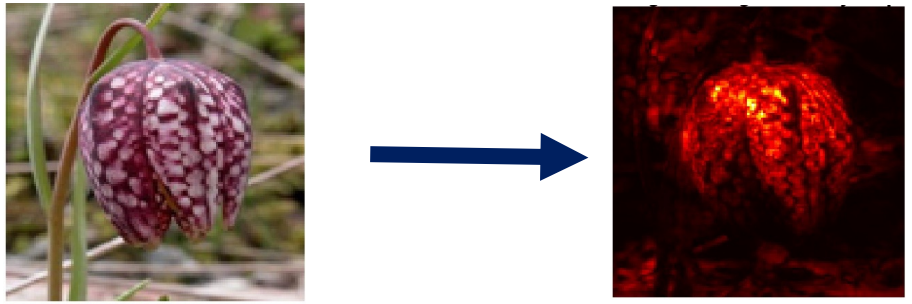
perturbed image,
normally trained model



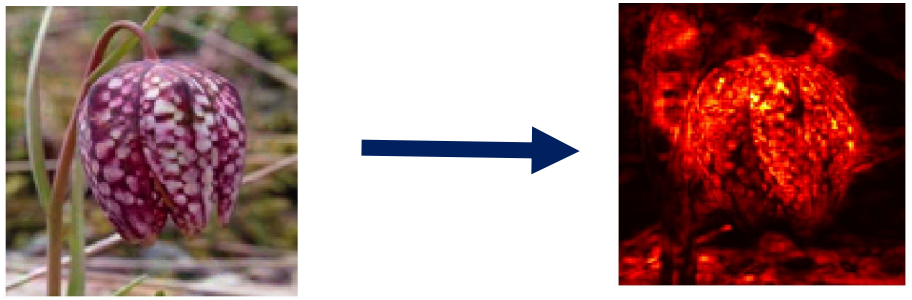
Robust Prediction Correlates with Robust Attribution: Why?

Empirical results demonstrate that if the model has robust prediction, usually it also has robust attribution.

original image,
robustly trained model



perturbed image,
robustly trained model



Adversarial Training (AT)

Training for robust prediction: find a model that predicts the same label for all perturbed inputs around the training input.

$$\min_{\theta} \mathbb{E}_{(x,y)\sim P}\left[\max_{x' \in \Delta(x)} \ell(x', y; \theta)\right]$$

Perturbed input

Allowed perturbations

Madry, Aleksander, et al. “Towards deep learning models resistant to adversarial attacks.” ICLR 2018

Solving AT Objective

We use projected gradient descent (PGD) to solve the inner maximization problem and then use stochastic gradient descent (SGD) to optimize the model parameters.

$$\min_{\theta} \mathbb{E}_{(\boldsymbol{x},y)\sim P}\left[\max_{\boldsymbol{x}'\in\Delta(\boldsymbol{x})} \ell(\boldsymbol{x}', y; \theta)\right]$$

$$\boldsymbol{x}^0 = \boldsymbol{x}$$
$$\boldsymbol{x}^{t+1} = \Pi_{\Delta(\boldsymbol{x})}(\boldsymbol{x}^t + \alpha \cdot sign(\nabla_{\boldsymbol{x}} \ell(\boldsymbol{x}^t, y; \theta)))$$
$$\boldsymbol{x}' = \boldsymbol{x}^T$$

PGD

$$\theta' = \theta - \eta \cdot \nabla_{\theta} \ell(\boldsymbol{x}', y; \theta)$$

SGD

Robust Attribution Regularization (RAR)

Training for robust attribution: find a model that can get similar attributions for all perturbed inputs around the training input.

$$\min_{\theta} \mathbb{E}_{(\boldsymbol{x},y)\sim P}[\ell(\boldsymbol{x}, y; \theta) + \lambda \cdot \mathrm{RAR}]$$

$$\mathrm{RAR} = \max_{\boldsymbol{x}' \in \Delta(\boldsymbol{x})} s(\mathrm{IG}(\boldsymbol{x}, \boldsymbol{x}'))$$

Perturbed input

Allowed perturbations

Chen, Jiefeng, et al. "Robust attribution regularization." *NeurIPS* 2019.

Robust Attribution Regularization (RAR)

Training for robust attribution: find a model that can get similar attributions for all perturbed inputs around the training input.

$$\min_{\theta} \mathbb{E}_{(\boldsymbol{x},y)\sim P}[\ell(\boldsymbol{x}, y; \theta) + \lambda \cdot \mathrm{RAR}]$$

$$\mathrm{RAR} = \max_{\boldsymbol{x}' \in \Delta(\boldsymbol{x})} s(\mathrm{IG}(\boldsymbol{x}, \boldsymbol{x}'))$$

Size function

Integrated Gradient

Chen, Jiefeng, et al. "Robust attribution regularization." *NeurIPS* 2019.

IG in RAR

- The IG function for RAR is defined as:

$$IG_i(x, x') \coloneqq (x_i' - x_i) \times \int_{\alpha=0}^{1} \frac{\partial \ell_y(x + \alpha \times (x' - x))}{\partial x_i'} d\alpha$$

where $\ell_y(x) = \ell(x, y; \theta)$ is the loss function. The input x is regarded as the baseline.

- From the axiom of Completeness, we have

$$\sum_{i=1}^{d} IG_i(x, x') = \ell(x', y; \theta) - \ell(x, y; \theta)$$

- In implementation, we use summation approximation of IG:

$$IG_i^{approx}(x, x') \coloneqq (x_i' - x_i) \times \sum_{k=1}^{m} \frac{\partial \ell_y \left(x + \frac{k}{m} \times (x' - x) \right)}{\partial x_i'} \times \frac{1}{m}$$

Connection to Robust Prediction

- Robust attribution regularization:

$$\min_{\theta} \mathbb{E}_{(\boldsymbol{x},y)\sim P}[\ell(\boldsymbol{x}, y; \theta) + \lambda * \mathrm{RAR}]$$

$$\mathrm{RAR} = \max_{\boldsymbol{x}' \in \Delta(\boldsymbol{x})} s(\mathrm{IG}(\boldsymbol{x}, \boldsymbol{x}'))$$

- If $\lambda = 1$ and $s(\cdot) = sum(\cdot)$, then RAR becomes the Adversarial Training objective for robust prediction:

$$\min_{\theta} \mathbb{E}_{(\boldsymbol{x},y)\sim P}\left[\max_{\boldsymbol{x}' \in N(\boldsymbol{x},\epsilon)} \ell(\boldsymbol{x}', y; \theta)\right]$$

simply by the Completeness of IG.

Madry, Aleksander, et al. “Towards deep learning models resistant to adversarial attacks.” ICLR 2018

When the two coincide?

Theorem. Consider the special case of one-layer neural networks, where the loss function takes the form of $\ell(\boldsymbol{x}, y; \boldsymbol{w}) = g(-y\langle \boldsymbol{w}, \boldsymbol{x} \rangle)$. Suppose g is nonnegative, differentiable, non-decreasing, and convex. Then for $\lambda = 1$, $s(\cdot) = \|\cdot\|_1$, and ℓ_∞ neighborhood, RAR training objective reduces to adversarial training objective:

$$\sum_{i=1}^{m} \max_{\|\boldsymbol{x}_i' - \boldsymbol{x}_i\|_\infty \le \epsilon} g(-y_i \langle \boldsymbol{w}, \boldsymbol{x}_i' \rangle) \qquad \text{(Adversarial training objective)}$$

$$= \sum_{i=1}^{m} g(-y_i \langle \boldsymbol{w}, \boldsymbol{x}_i \rangle + \epsilon \|\boldsymbol{w}\|_1) \qquad \text{(soft-margin)}$$

When the two coincide?

For the special case of one-layer neural networks (linear function), the robust attribution instantiation ($s(\cdot) = \|\cdot\|_1$) and the robust prediction instantiation ($s(\cdot) = sum(\cdot)$) coincide, and both reduce to soft max-margin training.

Connection to Robust Prediction

- Robust attribution regularization:

$$\min_{\theta} \mathbb{E}_{(\boldsymbol{x},y)\sim P}[\ell(\boldsymbol{x}, y; \theta) + \lambda * \mathrm{RAR}]$$

$$\mathrm{RAR} = \max_{\boldsymbol{x}' \in \Delta(\boldsymbol{x})} s(\mathrm{IG}(\boldsymbol{x}, \boldsymbol{x}'))$$

- If $\lambda = \lambda'/\epsilon^q$ and $s(\cdot) = \|\cdot\|_1^q$ with approximate IG, then RAR becomes the Input Gradient Regularization for robust prediction:

$$\min_{\theta} \mathbb{E}_{(\boldsymbol{x},y)\sim P}\left[\ell(\boldsymbol{x}, y; \theta) + \lambda' \|\nabla_{\boldsymbol{x}} \ell(\boldsymbol{x}, y; \theta)\|_q^q\right]$$

Ross, Andrew, and Finale Doshi-Velez. “Improving the adversarial robustness and interpretability of deep neural networks by regularizing their input gradients.” AAAI 2018.

Instantiations of RAR

- **IG-NORM**: if we pick $s(\cdot) = \|\cdot\|_1$, then this gives

$$\min_{\theta} \mathbb{E}_{(\boldsymbol{x},y)\sim P}\left[\ell(\boldsymbol{x}, y; \theta) + \lambda \cdot \max_{\boldsymbol{x}' \in N(\boldsymbol{x},\epsilon)} \|\mathrm{IG}(\boldsymbol{x}, \boldsymbol{x}')\|_1\right]$$

- **IG-SUM-NORM**: if we define $s(\cdot) = sum(\cdot) + \beta\|\cdot\|_1$, where $\beta \geq 0$ is a regularization parameter, and set $\lambda = 1$, then this gives

$$\min_{\theta} \mathbb{E}_{(\boldsymbol{x},y)\sim P}\left[\max_{\boldsymbol{x}' \in N(\boldsymbol{x},\epsilon)} \ell(\boldsymbol{x}', y; \theta) + \beta \cdot \|\mathrm{IG}(\boldsymbol{x}, \boldsymbol{x}')\|_1\right]$$

Solving RAR Objective

Given $(\boldsymbol{x}, y)$ at time step t during training, we have the following two steps:

1) **Attack step**: we run PGD on $(\boldsymbol{x}, y)$ to find $\boldsymbol{x}^*$ that produces a large inner max term (i.e., $\|\mathrm{IG}(\boldsymbol{x}, \boldsymbol{x}^*)\|_1$ for IG-NORM and $\ell(\boldsymbol{x}^*, y; \theta) + \beta \cdot \|\mathrm{IG}(\boldsymbol{x}, \boldsymbol{x}^*)\|_1$ for IG-SUM-NORM).
2) **Gradient step**: fixing $\boldsymbol{x}^*$, we can then compute the gradient of the corresponding loss with respect to θ, and then update the model.

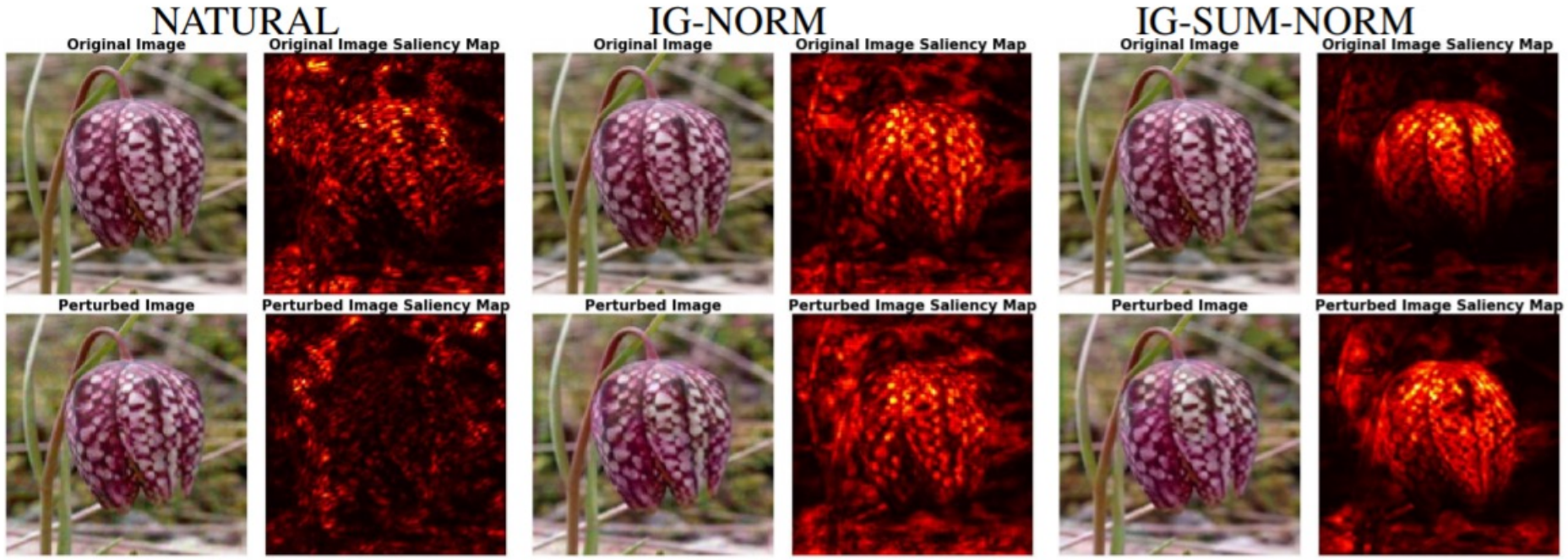
Difficulty of Optimization

Due to the summation approximation of IG, we have first order terms in the training objective. It forces us to compute second derivatives, which may not be numerically stable for deep networks.

$$\min_{\theta} \mathbb{E}_{(\boldsymbol{x},y)\sim P}\left[\ell(\boldsymbol{x},y;\theta) + \lambda \cdot \max_{\boldsymbol{x}'\in N(\boldsymbol{x},\epsilon)} \|\mathrm{IG}^{approx}(\boldsymbol{x},\boldsymbol{x}')\|_1\right]$$

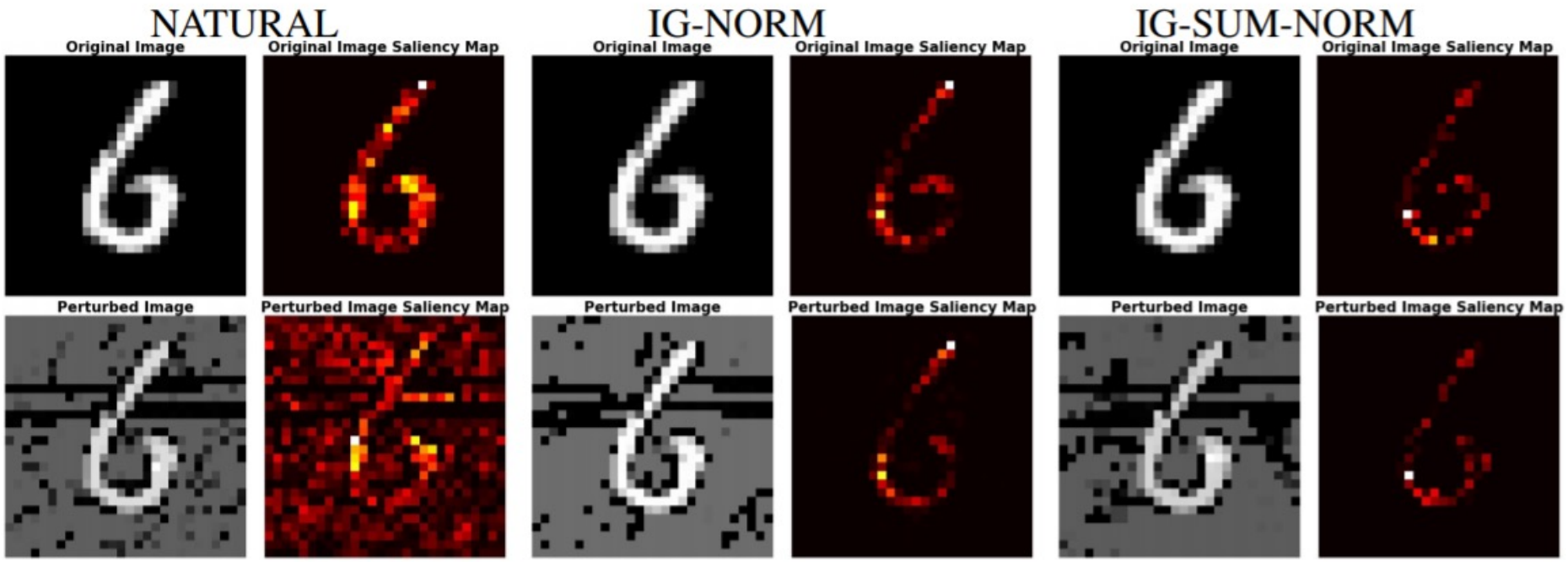
$$IG_i^{approx}(x,x') \coloneqq (x_i' - x_i)\times \sum_{k=1}^{m} \frac{\partial \ell_y\left(x + \frac{k}{m}\times(x'-x)\right)}{\partial x_i'} \times \frac{1}{m}$$

Experiments: Qualitative



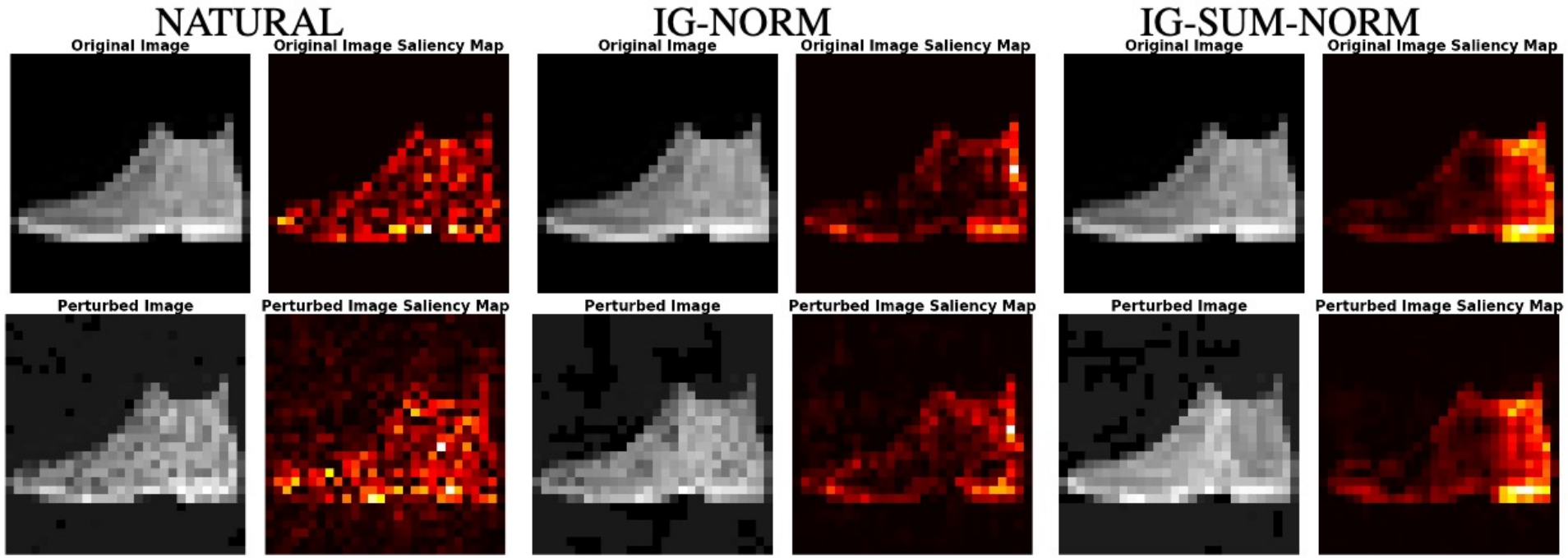
Flower dataset

Experiments: Qualitative



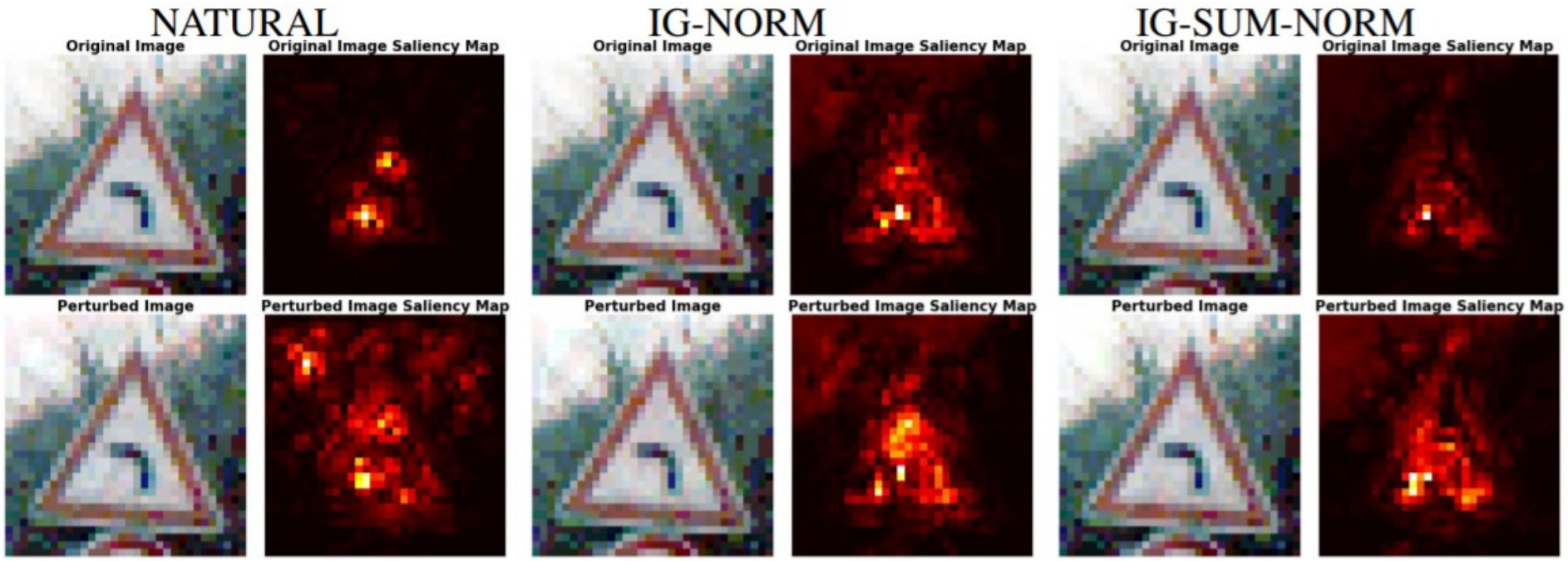
MNIST dataset

Experiments: Qualitative



Fashion-MNIST dataset

Experiments: Qualitative

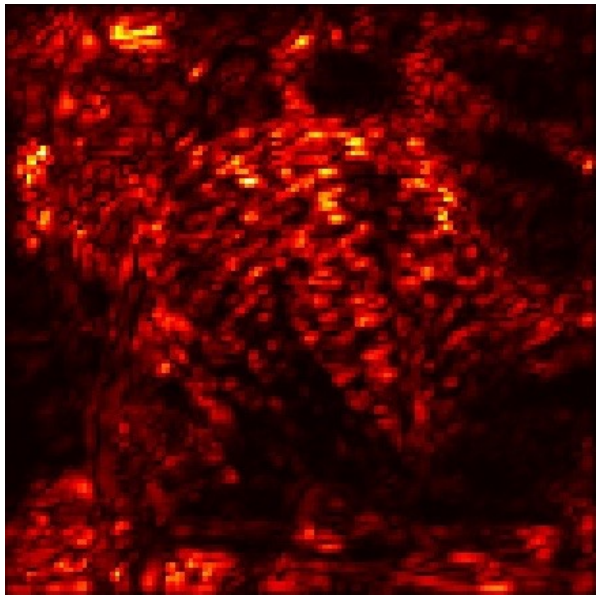


GTSRB dataset

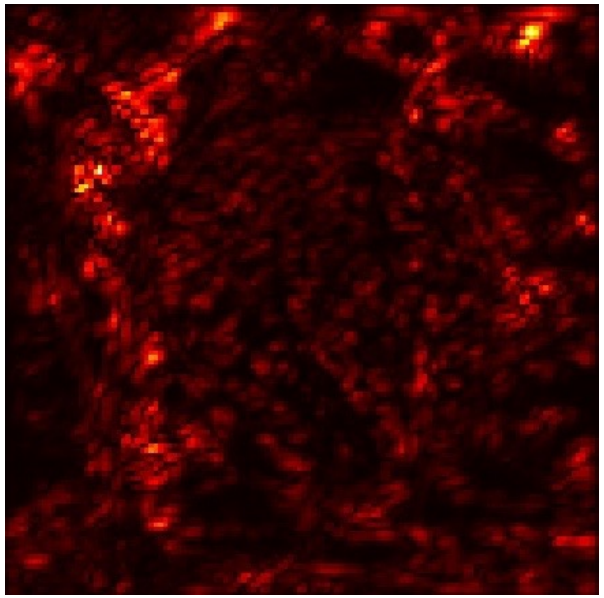
Experiments: Quantitative

- Metrics for attribution robustness:
  1. Kendall's tau rank order correlation.
  2. Top-K intersection.

Original Image Attribution Map



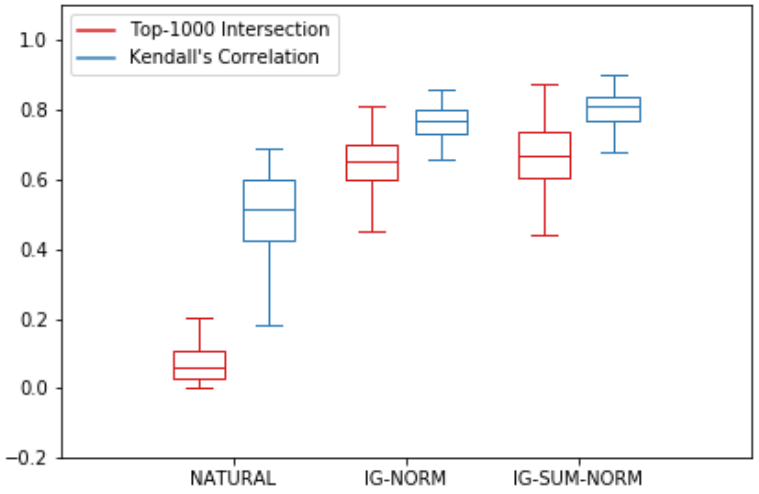
Perturbed Image Attribution Map



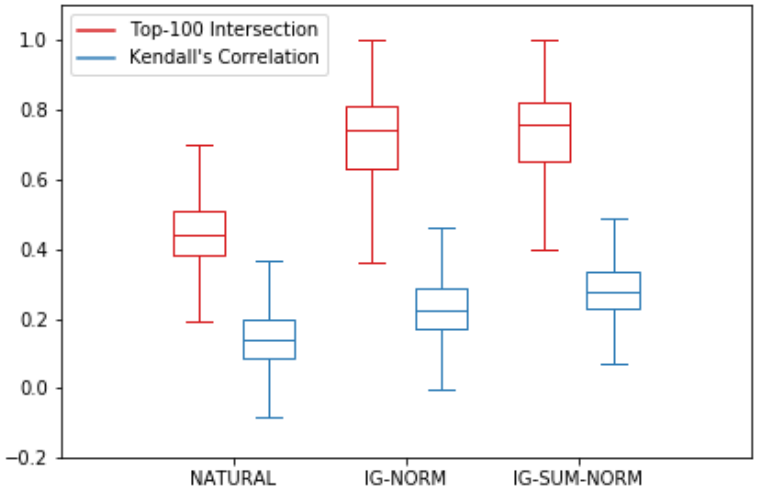
Top-1000 Intersection: 0.1%
Kendall's Correlation: 0.2607

Experiments: Quantitative

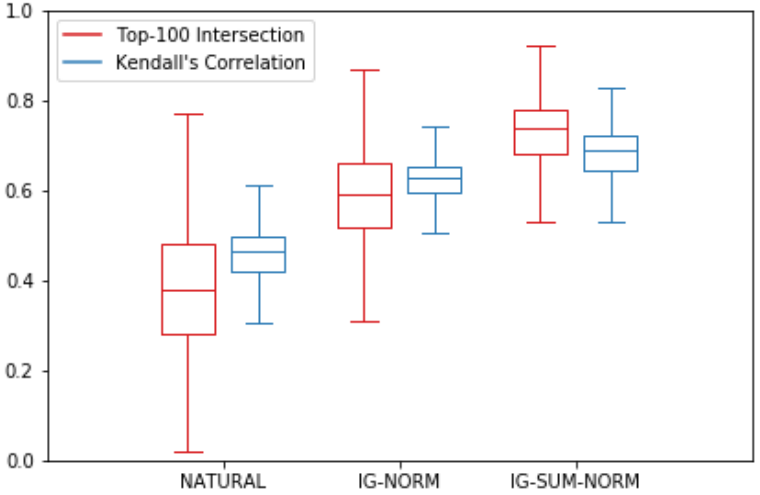
Flower



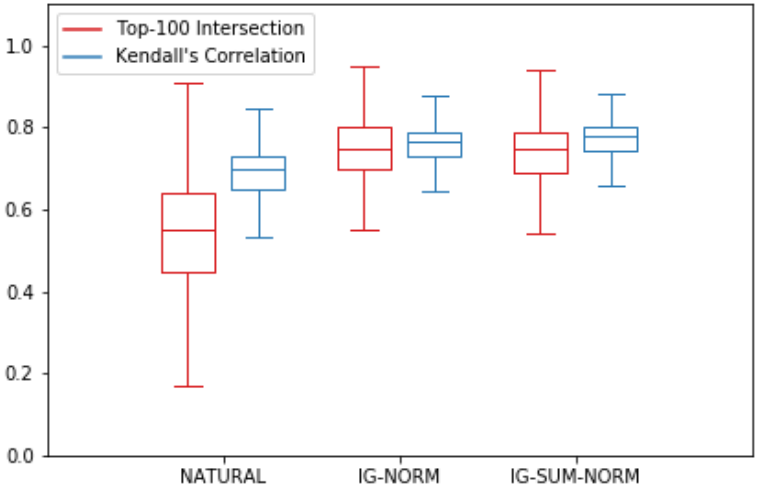
MNIST



Fashion-MINST



GTSRB



Prediction Accuracy of Different Models

Dataset	Approach	Nat. Acc.	Adv. Acc.
MNIST	NATURAL	99.17%	0.00%
	AT	98.40%	92.47%
	IG-NORM	98.74%	81.43%
	IG-SUM-NORM	98.34%	88.17%
Fashion-MNIST	NATURAL	90.86%	0.01%
	AT	85.73%	73.01%
	IG-NORM	85.13%	65.95%
	IG-SUM-NORM	85.44%	70.26%
GTSRB	NATURAL	98.57%	21.05%
	AT	97.59%	83.24%
	IG-NORM	97.02%	75.24%
	IG-SUM-NORM	95.68%	77.12%
Flower	NATURAL	86.76%	0.00%
	AT	83.82%	41.91%
	IG-NORM	85.29%	24.26%
	IG-SUM-NORM	82.35%	47.06%

Empirical Observations

Our main findings can be summarized as follows:

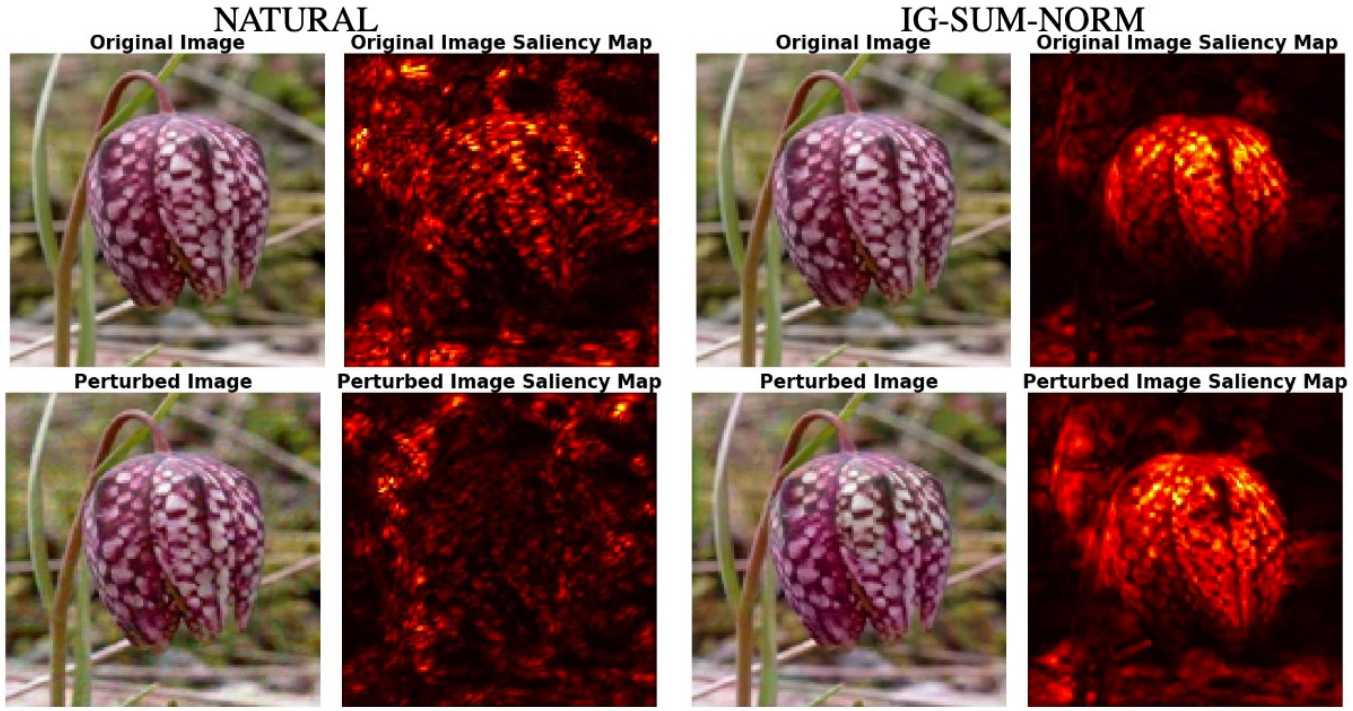
1. Compared with naturally trained models, RAR only results in a very small drop in test accuracy;
2. Our method gives significantly better attribution robustness, as measured by correlation analyses;
3. Our models yield comparable prediction robustness (sometimes even better), compared with adversarially trained models (for robust prediction), while consistently improving attribution robustness;
4. Intriguingly, RAR leads to much more human aligned attribution.

Overview

- What is attribution?
- How to compute attribution?
- What is the vulnerability issue of attribution methods?
- Why should attribution methods be robust?
- How to enforce attribution robustness?
- What are the benefits of attribution robustness?

Benefits of Attribution Robustness

- Robust attribution correlates with robust prediction.
- Robust attribution leads to more human-aligned attribution.
- Robust attribution may help tackle spurious correlations.



Top-1000 Intersection: 0.1%
Kendall's Correlation: 0.2607

Top-1000 Intersection: 60.1%
Kendall's Correlation: 0.6951

Question?

Reference

- Sundararajan, Mukund, Ankur Taly, and Qiqi Yan. "Axiomatic attribution for deep networks." International conference on machine learning. PMLR, 2017.
- Ghorbani, Amirata, Abubakar Abid, and James Zou. "Interpretation of neural networks is fragile." Proceedings of the AAAI conference on artificial intelligence. Vol. 33. No. 01. 2019.
- Madry, Aleksander, et al. "Towards Deep Learning Models Resistant to Adversarial Attacks." International Conference on Learning Representations. 2018.
- Chen, Jiefeng, et al. "Robust attribution regularization." Advances in Neural Information Processing Systems 32 (2019).