

# A Unified Study of Machine Learning Explanation Evaluation Metrics

Yipei Wang, Xiaoqian Wang\*  
 Elmore School of Electrical and Computer Engineering  
 Purdue University  
 West Lafayette, IN 47907  
 wang4865@purdue.edu, joywang@purdue.edu

## Abstract

The growing need for trustworthy machine learning has led to the blossom of interpretability research. Numerous explanation methods have been developed to serve this purpose. However, these methods are deficiently and inappropriately evaluated. Many existing metrics for explanations are introduced by researchers as by-products of their proposed explanation techniques to demonstrate the advantages of their methods. Although widely used, they are more or less accused of problems. We claim that the lack of acknowledged and justified metrics results in chaos in benchmarking these explanation methods – *Do we really have good/bad explanation when a metric gives a high/low score?* We split existing metrics into two categories and demonstrate that they are insufficient to properly evaluate explanations for multiple reasons. We propose guidelines in dealing with the problems in evaluating machine learning explanation and encourage researchers to carefully deal with these problems when developing explanation techniques and metrics.

## 1 Introduction

The rapid development of deep neural networks (DNNs) has achieved great success, leading to widespread applications in reality. However, the applications of such black boxes in high-stake areas have drawn more and more concerns. General Data Protection Regulation (GDPR) stipulates a right to explain algorithmic decision making. Due to the growing needs, research in Explainable Artificial Intelligence (XAI) – especially explainable machine learning, has been a hit in recent years. As a consequence, a rich genre of explanation methods have been proposed, serving the purposes of exploring and studying the inner mechanism of black-box machine

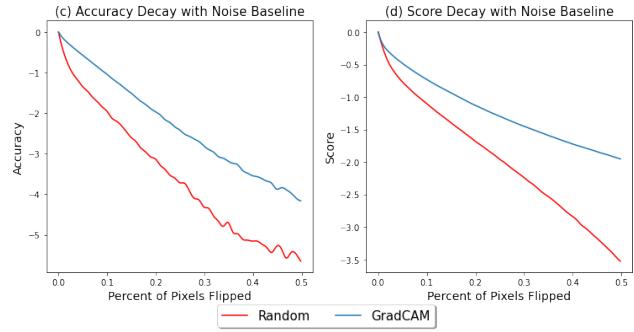


Figure 1: Results of using Pixel Flipping metric (Samek et al., 2016) to evaluate two explanations (GradCAM (Selvaraju et al., 2017) explanation and Random explanation) of VGG-16 model on ImageNet. Pixels are masked in descending order of attribution values. Random masking yields faster decay than GradCAM, which according to Pixel Flipping metric, shows that Random explanation captures “more important” pixels than GradCAM in explanation.

learning models.

Despite the prosperity in explanation methods, recent studies show that a nonnegligible portion of such explanation methods have severe intrinsic flaws (Adebayo et al., 2018; Srinivas and Fleuret, 2020; Shah et al., 2021). Besides, even established on reasonable criteria, different explanation methods provide different explanations for the same model-input pair, which can be confusing and undermine the trust from end users, because it is hard to decide *which* explanation to trust if they are contradictory. This chaos makes finding the proper explanation method as difficult as understanding a black-box model for end users – The users have to actually know the mechanism of the black-box model to find the desired explanation method, which is a vicious circle. Therefore, metrics that are used to evaluate explanation methods have come into play.

Based on the difference in measured statistics, quantitative metrics for explanations can be roughly divided into two

\*corresponding author

categories. The first category is the *alignment* metric. One can check if the explanations correspond to the prior knowledge. The other category is the *performance* metric, where the input samples are usually modified with respect to the explanations in some manners and then fed to the prediction model. The change of performance of the prediction model is used to evaluate the corresponding explanation method. These two categories of metrics have been widely used by researchers in both developing explanation methods and metrics. However, even though some problems of evaluating explanation methods with the different metrics have been noticed elsewhere, they have never been paid sufficient attention to. The lack of in-depth study on issues with using existing explanation metrics can be detrimental, and lead to chaotic situations in the entire XAI area.

Here we take one of the most widely used performance metrics, Pixel Flipping (Samek et al., 2016), as an example. In this metric, we mask pixels of input images according to attribution values based on an explanation method and then test performance decay as more pixels are masked. A good explanation method will ideally result in faster performance decay when pixels are masked in the descending order of attribution values. However, as shown in Figure 1, Random attribution (which assigns attribution values randomly) has faster decay than GradCAM (Selvaraju et al., 2017) in terms of both the log ratio of accuracy and predicted scores. But this by no means implies that Random attribution is a better explanation method than Grad-CAM. As a result, in this paper, we propose to ask the question, “*Do we really have a good/bad explanation when an evaluation metric says so?*”. In order to answer this question, we explore the reasons behind this chaos and carry out a unified study on the evaluation metrics for explanations.

Our contribution can be summarized as follows:

- We categorize existing metrics for attribution methods into *alignment* metrics and *performance* metrics and carry out various experiments in case studies to demonstrate the intrinsic flaws of both categories.
- We demonstrate that incompatible goals between explanation methods and metrics lead to unfair evaluation.
- We use *faithfulness* and *plausibility* to analyze the reasons behind the chaos of explanation evaluation. We introduce *projected gradient descent (PGD) enhancement* to illustrate the conflict between such two aspects.
- We propose criteria that guide how to avoid being misled by the chaos of explanation evaluation metrics.

## 2 Related Work

In this section we review related works that develop or evaluate explanation/interpretation in machine learning. We

clarify that the difference between notions *explainability* and *interpretability* is beyond the scope of this work. They are not distinguished and will be used interchangeably in the following context.

**Explanation Methods** Depending on the ways explanations are expressed, explanation methods can be divided into attribution methods and high-level methods. Given input data, attribution methods assign each feature (pixel in images, token in text, etc.) of the input a value, representing the importance or relevance of the feature to the output. Based on the mechanism of producing attribution explanations, there are back-propagation methods (Simonyan et al., 2013; Zeiler and Fergus, 2014; Springenberg et al., 2014; Bach et al., 2015; Zhou et al., 2016; Selvaraju et al., 2017; Sundararajan et al., 2017; Shrikumar et al., 2017; Zhang et al., 2018; Parekh et al., 2020), etc. and perturbation methods (Petsiuk et al., 2018; Lundberg and Lee, 2017; Fong et al., 2019), etc. Differently, high-level methods provide explanations for the prediction process from higher levels, such as concept-based explanations (Ribeiro et al., 2016; Kim et al., 2018; Zhou et al., 2018; Ghorbani et al., 2019; Koh et al., 2020), sample-based explanations (Koh and Liang, 2017; Chen et al., 2020), etc. Due to the space limit, we refer to Appendix A for other ways of characterizing explanation methods.

Among the different forms of explanation methods, attribution methods are more universal and have always been evaluated quantitatively together using the same metric (Petsiuk et al., 2018; Fong et al., 2019). While for other forms, a unified comparison can be difficult. For example, different concept-based methods may have different sets of concepts or even be developed for different datasets/tasks. Thus in this paper, we focus on metrics for attribution methods.

**Metrics for Evaluating Attribution Methods** Depending on the statistics measured by the metrics, attribution metrics can be divided into *alignment* metrics and *performance* metrics. Alignment metrics measure how well the explanation aligns with prior knowledge or given supervision information. Selvaraju et al. (2017) use weakly-supervised localization to evaluate the explanations. Following a similar idea, Zhang et al. (2018) simplify it by introducing Pointing Game, which calculates the ratio of the number of samples whose attribution maps can correctly point at the pre-annotated object area. Poerner et al. (2018); Yang and Kim (2019); Adebayo et al. (2020); Zhou et al. (2021) forge new datasets trying to introduce artificial ground truth.

In contrast, performance metrics measure the change in prediction model performance w.r.t. certain modification in input samples. Samek et al. (2016) introduce Pixel Flipping, which calculates the performance change when input features are perturbed based on the explanations. Most performance metrics share the same idea as Pixel Flipping and

resemble it. They modify the input according to the explanations and observe the change in predictions, such as the insertion/deletion metric (Petsiuk et al., 2018), masking top pixels (Chen et al., 2018b), top- $k$  ablation (Sturmels et al., 2020), Impact Score (Lin et al., 2019), word deletion (Arras et al., 2016), infidelity (Yeh et al., 2019), etc. Hooker et al. (2018) introduce ROAR, which is a modification of Pixel Flipping and requires training a completely new black-box model every time the number of masked pixels changes.

**Guidelines** Preece et al. (2018) argue that different stakeholder communities have different expectations of machine learning explanations. Miller (2019) carry out an analysis from the social scientific perspective. Jacovi and Goldberg (2020) propose guidelines in evaluating explanations according to faithfulness and plausibility. Tomsett et al. (2020) perform sanity checks on saliency metrics, which studies the properties of metrics by introducing the reliability from the psychometric testing.

### 3 Incompatible Goals

Denote by  $f : \mathbb{R}^d \rightarrow \mathbb{R}^c$  a prediction model before the last activation, where  $d$  is the dimension of input data,  $c$  is the dimension of the output. Then the attribution method for  $f$  is defined as  $\phi_f^i : \mathbb{R}^d \rightarrow \mathbb{R}^d$ . (For RGB images the codomain of  $\phi$  should be  $\mathbb{R}^{d/\text{channels}}$ , but here we omit this difference.)  $i \in [c]$  corresponds to the  $i$ -th class, and we will use  $t$  when referring to the true class. Since all attribution methods share the same form, it may seem natural to evaluate them altogether, with some metric  $M$ . However, we argue that different attribution methods may focus on different goals, and hence should not always be evaluated together.

For example, despite it is preferred by many researchers that an explanation method should satisfy  $\phi_f^i(\mathbf{x}) = f(\mathbf{x})_i, \forall i \in [c]$  (Bach et al., 2015; Sundararajan et al., 2017; Lundberg and Lee, 2017), which we refer to as *completeness*, a lot of attribution methods do not satisfy this property. But *should they?* The Gradient method does not satisfy completeness, but it does not focus on reassigning  $f(\mathbf{x})_i$  to the input in the first place. For methods where attribution values are *additive contributions* to the predictions, this is a reasonable property, but the attribution values from the gradient method only represent the *sensitivity* of the prediction  $f(\mathbf{x})_i$  w.r.t. the corresponding input features in their neighborhoods. Also, suppose  $f$  has only one layer and degenerates to a linear model such that  $f(\mathbf{x}) = W^T \mathbf{x}$ , where  $W \in \mathbb{R}^{d \times c}$ . Then  $\forall i \in [c]$ , if we set  $w_j^{(i)} x_j$  as the attribution value for  $x_j$  as the explanation, it satisfies completeness. However, the gradient  $\nabla_{\mathbf{x}} f(\mathbf{x})_i = \mathbf{w}^{(i)}$  does not. But this explanation is as important as the complete explanation  $w_j^{(i)} x_j$ . Therefore, one has to be aware of the goals of both the explanation methods and the metrics. A metric can reasonably evaluate

Table 1: Model-Explanation-Metric triple combinations. The bottom row shows metric evaluation results, provided the metric measures *plausibility* of explanation. “✓” indicates high metric score, and “✗” indicates low score.

Model $f$	Plausible		Implausible	
	Faithful	Unfaithful	Faithful	Unfaithful
Explanation $\phi$	✓	✗	✗	Uncertain
Metric $M$				

an explanation only if their goals are compatible.

### 4 The Pitfall of Plausibility

There are two main bodies in XAI research, machines and humans. The goal of XAI is to convey information from machines to humans, in a comprehensible way. Machine wise, the *faithfulness* of explanation refers to that explanation reflects the true mechanism of the prediction model  $f$ . While human wise, *plausibility* refers to that explanation conforms with humans’ perception. It may seem natural to expect explanations that are both faithful and plausible. However, these two aspects can be contradictory and easily confused in the development and evaluation of explanations.

Many existing explanation metrics – including all alignment metrics, lean towards plausibility. They evaluate explanations by checking if the explanations are compatible with some prior knowledge. Such prior knowledge is *independent* of model  $f$ . This is a tempting yet dangerous way, because the true mechanism of model  $f$  is not necessarily plausible, and may not correspond to the prior knowledge humans possess.

On the one hand, a model  $f$  can make predictions in a way that is completely different from human knowledge, such as textures (Geirhos et al., 2018), controlled ground truth (Kim et al., 2018), overinterpretations (Carter et al., 2020), perturbations by the adversarial attack, overfitting features, etc. Although they are all recognized to have a great influence on the mechanism of the black-box models, none of them are plausible to humans, and none of them can be easily included in the prior knowledge. On the other hand, a plausible “explanation” can have nothing to do with the predictive mechanism of the black-box model, such as edge detectors (Adebayo et al., 2018).

Therefore, metrics that focus on plausibility will not provide useful information about the inner mechanism. Even worse, it can jeopardize the evaluation of explanations. As shown in Table 1, when the mechanism of the model itself is plausible (e.g. focusing on objects when classifying), a *faithful* explanation will also be plausible. Hence this flaw is not very evident. However, when the model is implausible, a *faithful*

explanation will capture the implausibility and is thereby implausible. In this case, it will be unfairly evaluated by plausibility metrics.

## 5 Projected Gradient Descent Enhancement

In order to demonstrate how plausibility can undermine faithfulness, here we propose projected gradient descent (PGD) enhancement, which is the inverse of PGD attack (Madry et al., 2017). PGD enhancement uses exactly the same criterion as PGD attack, but changes the direction of each step. As a result, given a randomly initialized, untrained neural network, it produces enhanced samples with human-imperceptible perturbations. Such enhanced samples can be correctly classified by the untrained model. We test AlexNet (Krizhevsky, 2014), VGG-16 (Simonyan and Zisserman, 2014), and MnasNet (Tan et al., 2019) on the validation set of ISLVRC 2012 (Deng et al., 2009). Please refer to Appendix B for implementation details.

As shown in Table 2, given an enhanced validation set, an untrained model can have comparable performance to a well-trained model (trained and tested on the raw datasets). Besides, the desired explanation method should give them very different explanations since they have intrinsically different mechanisms (trained and untrained).

We take VGG-16 and Grad-CAM as an example and plot the explanation heatmaps in Figure 2. Please refer to Appendix B for other explanation methods. In Figure 2, there is no doubt that explanations in the bottom row are the most plausible ones among rows. The objects are highlighted precisely, and intuitively, very little uninformative area is highlighted. However, if evaluated in the sense of plausible explanation, one will justify that Grad-CAM is bad based on the second row. This judgement is neither fair nor reasonable since it may correspond to the third case in Table 1, where the model itself is implausible while the explanation is faithful to this implausibility. Then a plausibility-based metric will unfairly evaluate the explanation.

This example demonstrates the reason why plausibility should be used carefully in metrics to evaluate explanations. Explanations should first be faithful and can reflect the true mechanism of the model. In Figure 2, the differences in mechanism between the middle and the bottom row should be ascribed to two factors: 1) the differences between the trained and the untrained VGG-16 networks; and 2) the human-imperceptible PGD enhancement. However, neither the level of training in the model nor the enhancement in data is plausible to humans. As a consequence, we suggest that plausibility should be specifically separated from faithfulness in evaluations, and both aspects should be carefully considered in both developing explanation methods and evaluation metrics. Otherwise, the chaos caused by the

Table 2: Accuracy comparison between the trained models on raw data and the untrained models on PGD-enhanced data. Three different model structures (AlexNet, VGG-16, MnasNet) are used.

		Raw	Enhanced
AlexNet	Untrained	0.086%	<b>64.664%</b>
	Trained	52.558%	-
VGG-16	Untrained	0.102%	<b>99.956%</b>
	Trained	72.230%	-
MnasNet	Untrained	0.110%	64.588%
	Trained	<b>71.718%</b>	-

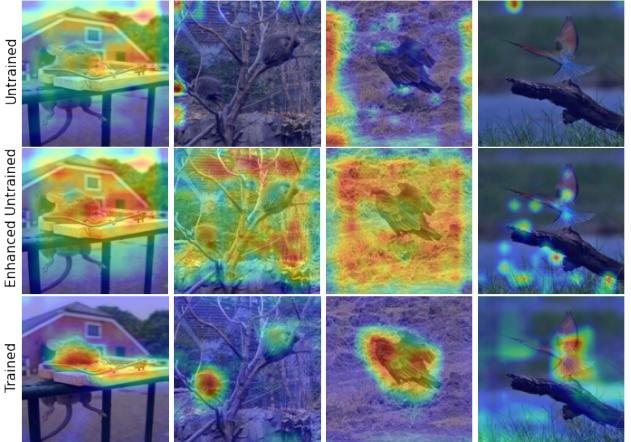


Figure 2: Grad-CAM explanations of VGG-16 models. The results are from the untrained model with raw data as input (top row), the untrained model with PGD-enhanced data as input (middle row), and the trained model with raw data as input (bottom row). All samples of the middle and the bottom rows are correctly classified, and all samples from the top row are falsely classified.

confusion of faithfulness and plausibility will impede the interpretability research. We further look into the chaos in explanation evaluation in the following case studies.

## 6 Case Studies

Among various explanation metrics, we take one representative and widely used metric in each category for the case study and demonstrate the chaos. In the demonstration, we use sparse linear models with Lasso regularization as the benchmark for *faithfulness* in all experiments. The reasons for choosing such a benchmark include two folds: 1) sparse linear model is an attribution method; 2) white-box models like sparse linear models can axiomatically reflect the true mechanism of itself while building a truly faithful post-hoc explanation for DNNs still remains an open question. We

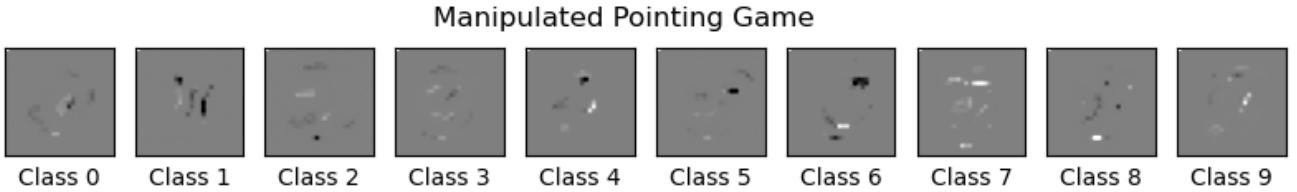


Figure 3: The explanation  $\hat{\mathbf{w}}_i \odot \mathbf{x}$  of the manipulated sparse linear model for a digit 7 from MNIST dataset. We constrain the sparse linear model such that the top left corner pixel has the highest attribution score. The explanation from such model is faithful but has very low Pointing Game ratio.

use the Hadamard product  $\hat{\mathbf{w}}^{(i)} = \mathbf{w}^{(i)} \odot \mathbf{x} \in \mathbb{R}^d, i \in [c]$  as the attribution explanation of the sparse linear model.

## 6.1 Alignment Metrics

Among alignment metrics that measure how well explanations conform with prior knowledge, we take Pointing Game (Zhang et al., 2018) as a representative. Pointing Game is one of the most popular metrics that measure attribution explanation methods. Please refer to Appendix C for more details of this metric. Here, the annotated bounding boxes (or the object segmentation) are the prior knowledge. This prior knowledge will directly result in a plausibility metric since the annotations are solely based on the way we humans perceive the world – the objects. Intuitively, pixels that contribute highly to the prediction of the targeted classes should be on the objects, or at least be in the corresponding bounding boxes. However, if a prediction of a model  $f$  is based on human-imperceptible features, (which, as discussed earlier, is not a rare scenario) then Pointing Game fails to properly evaluate the correctness of an explanation  $\phi_f$ . Besides, if we have  $\phi_f \equiv \phi$  independent from  $f$  to be an edge detector, it will gain a high score in Pointing Game, but has no meaning in explaining the model  $f$ .

Here we take the sparse linear model as an example since it is axiomatically acknowledged as faithfully interpretable. However, such a sparse linear model can be too simple for complex image datasets with annotations, such as Pascal VOC (Everingham et al., 2010), COCO (Lin et al., 2014), etc. Therefore, we build an annotated MNIST dataset with bounding boxes. Please refer to Appendix D for the illustration of bounding boxes for some examples. It is expected by Pointing Game that the pixel with the highest attribution value should be in the bounded area. This is very intuitive and tempting. However, it can be easily demonstrated that this metric is not consistent with the true mechanism of  $f$ . For sparse linear models, the true (additive) mechanism is the Hadamard product  $\hat{\mathbf{w}}^{(t)} \odot \mathbf{x}$ . We modify the weights corresponding to the top left corner of all classes, so that the top left pixel has the largest attribution values in all classes. In this way, on the one hand, the most contributing pixel in

Table 3: Pointing Game ratio results.

	Standard Sparse Linear Model	Manipulated Sparse Linear Model
Accuracy	79.51%	79.51%
Pointing Game Ratio	100.0%	0.0%

Pointing Game will always point to the top left corner. On the other hand, since the manipulation raises the prediction scores of all classes by the same amount, the classification results stay the same. Besides, here only one weight is changed from zero to non-zero, and the sparsity is preserved, so the model is still a sparse linear model, and it is still recognized as a self-interpretable model.

We show an example of explanations of input digit 7 in Figure 3. It can be found that the left corner has the highest attribution value in the classification process, but does not conform with the prior knowledge in Pointing Game. The comparison results are shown in Table 3. We can find that Pointing Game gives very delusive results. Through such minor and imperceptible change, the sparse linear model preserves all properties including the expressiveness (same accuracy), but the Pointing Game score drops from 100% to 0. That is, a human-imperceptible change in model explanation can make a drastic change in the Pointing Game ratio, which means it is also very sensitive.

As a representative of alignment metrics, the above analysis on Pointing Game can be easily generalized to other metrics of this category. We argue that the mismatch between the explanations and the prior knowledge should be ascribed to the model  $f$ , instead of the explanation  $\phi_f$ . On the contrary, the obligation of explanations is indeed to find the *mismatch* between  $f$  and prior knowledge. In this way, the explanations enable us to debug the model  $f$ , and build more transferable models.

## 6.2 Performance Metrics

Among performance metrics that measure changes in model performance w.r.t. certain modifications in input samples,

Pixel Flipping (Samek et al., 2016) is a typical representative in this category. Other members such as word deletion (Arras et al., 2016), masking top pixels (Chen et al., 2018b), the deletion/insertion metric (Petsiuk et al., 2018), top- $k$  ablation (Sturmels et al., 2020), etc. can be seen as variations of Pixel Flipping. All of these variations are based on the same criteria, that is, to mask (or insert for the insertion metrics) pixels by the order of their importance scores assigned by the explanation method. Usually, the change of prediction score  $f(\mathbf{x})_t$  (and change of accuracy in other versions) are measured with respect to the portion of masked pixels in image data (or other kinds of features in other data types). For clarity, we will focus on Pixel Flipping in the following context. It is also worth emphasizing that the flaws of Pixel Flipping are shared by other performance metrics.

Given an attribution method  $\phi_f^t(\mathbf{x})$ , all  $d$  features can be ranked by an index set  $A = \{\alpha_1, \alpha_2, \dots, \alpha_d\}$ , which is a permutation of  $[d]$ . Pixel Flipping masks pixels by the order of the index set  $A$ . That is,  $x_{\alpha_1}, x_{\alpha_2}, \dots, x_{\alpha_d}$ . Generally, the index set  $A$  is built by the monotonically decreasing order, such that  $\phi_f^t(\mathbf{x})_{\alpha_1} \geq \phi_f^t(\mathbf{x})_{\alpha_2} \geq \dots \geq \phi_f^t(\mathbf{x})_{\alpha_d}$ , where a more important pixel decided by the explanation method will be masked earlier. Therefore, when evaluating  $\phi_f$ , a faster decay on the prediction score (or the accuracy) is more desired. It can be noticed that Pixel Flipping aims at capturing the inner mechanism of  $f$ . It achieves this by making the decoupling assumptions on input features. This is a very strong assumption and is only true for linear additive models. As a consequence, the flaws of Pixel Flipping are from multiple aspects.

### 6.2.1 THE INDICATOR FLAW

Due to the independence assumption of Pixel Flipping, we know that if  $f$  is a linear model, then the assumption holds, and that in this case Pixel Flipping is faithful. However, we argue that the indicator should be the change in predicted score instead of accuracy for Pixel Flipping. If the accuracy is used as the indicator, Pixel Flipping fails even for linear models. This is actually from the formulation of attribution methods. No matter based on backpropagation or perturbation, attribution methods are always subject to some selected target  $f(\mathbf{x})_i, i \in [c]$ , which does not directly affects the accuracy. The accuracy is determined by the final predicted result  $t' = \arg \max_{i \in [c]} f(\mathbf{x})_i$ , which contains the information of all classes. Now that  $\phi_f^t$  itself does not contain information about the predicted results of classes  $\forall j, j \neq t$ , using accuracy as the indicator is an ill-posed way to evaluate  $\phi_f$ . It is not guaranteed that masking the pixel with the truly highest contribution will result in the fastest decay in accuracy, because an input feature being the *most important* to the true class  $t$  does not mean that it is not as important to the class  $j$ . Please refer to Appendix E for the formal statement and

the construction of the counterexample.

### 6.2.2 THE GOALS FLAW

As mentioned in Section 3, the goal of the metric  $M$  has to be compatible with the goal of the attribution method  $\phi_f$  to properly evaluate the attribution method. For Pixel Flipping, the goal is similar to completeness. If  $\phi_f$  represents sensitivity, relevance, or basically any property other than the additive contribution, it is not appropriate to evaluate them with Pixel Flipping. We still take  $f$  to be linear as an example, since DNNs will introduce too much exogenous bias as we explain later. Let  $c = 1, f(\mathbf{x}) = \mathbf{w}^T \mathbf{x}$ . If we use the Hadamard product as the explanation, where the attribution value for  $x_j$  is  $w_j x_j$ , then  $\phi_f$  has the same goal as Pixel Flipping – to assign the additive contribution. However, if we use the weight  $w_j$  as the attribution value for  $x_j$ , its goal is no longer the same as Pixel Flipping. The attribution value  $w_j$  indicates how *changing*  $x_j$  influences the prediction. And correspondingly, the Hadamard product explanation aces Pixel Flipping, because if the attribution value  $w_j x_j$  is largest among unmasked values, then masking  $x_j$  results in the fastest decay in the performance. However, in the weight explanation, masking  $x_j$  with the largest attribution value  $w_j$  does not guarantee the largest decay. This demonstrates the goals flaw of Pixel Flipping.

### 6.2.3 THE DISTRIBUTION FLAW

One of the most fatal flaws of Pixel Flipping is the assumption of distributions. It implicitly assumes the independence among input features, which can be a too strong assumption for complex datasets or complex models. It is also argued that since  $f$  is trained on the original dataset, it is unreasonable to test its performance on the masked dataset since they have different distributions (Hooker et al., 2018).

We quantitatively demonstrate the out-of-distribution flaw of Pixel Flipping by introducing a random noise case in the comparisons among attribution methods. The experiments are carried out on the ILSVRC 2012 validation dataset. We use Pixel Flipping to measure several popular explanation methods<sup>1</sup>, including Grad-CAM (Selvaraju et al., 2017), Gradient (Simonyan et al., 2013), DeConvNet (Zeiler and Fergus, 2014), Excitation Back-propagation (Zhang et al., 2018), Guided Back-propagation (Springenberg et al., 2014), and Linear Approximation. Also, we include the random masking criterion as the benchmark, where the pixels are masked at uniform random. Images are resized to  $224 \times 224$ , and in the illustration,  $9k (\approx 18\%)$  pixels are masked for

---

<sup>1</sup>We clarify that due to the goals flaw, we disagree that some of these methods should be evaluated by performance metrics. The reason we include these methods in the evaluation here is not to show which explanation technique is better, but to illustrate the pitfalls of Pixel Flipping as an evaluation metric.

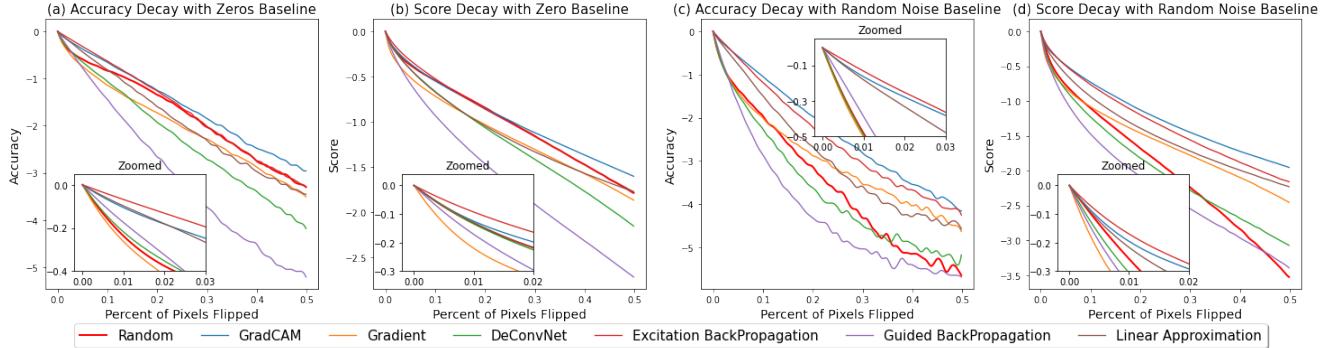


Figure 4: Pixel Flipping results with the zeros baseline (left two subfigures) and the random noise baseline (right two subfigures). All results are normalized to  $[0, 1]$  and then taken the logarithm for better illustration.

each sample. The measures are the predicted scores  $f(\mathbf{x})_t$ .

According to Pixel Flipping, the faster  $f(\mathbf{x})_t$  decays, the better  $\phi_f$  is. It is intuitively desired by anyone that random masking will have the worst performance, i.e., the slowest decay. However, as shown in Figure 4, we can find the counterintuitive results. At the beginning, random masking has a quite fast decay compared with explanation-based masking. As the portion of masked pixels increases, this phenomenon gradually diminishes.

Another very interesting observation from the experiment results in Figure 4 is that, if we pay attention to the explanation methods being tested, we can find that Guided BP, DeConvNet, and Gradient always outperform others. These three methods share a very important similarity with random masking – Their masked pixels are isolated, while the other three methods’ masked pixels always adjoin. Please refer to Appendix B for some illustrations. In addition to the “informative” pixels themselves, the distribution difference among explanation methods also has great impact on performance decay.

Now that both the out-of-distribution issue (pixel isolation) and the information of pixels contribute to the performance decay in Pixel Flipping, it is worth exploring how they interact and influence the performance decay. To compare the random masking benchmark with attribution-based masking more fairly, we carry out a Reference Pixel Flipping experiment. The  $N$  reference pixels are first selected based on the attribution method, then  $n (< N)$  pixels are randomly masked within the  $N$  reference pixels. The illustration of such masking is shown in Figure 5. In this way, by confining the random masked area as similar to the Grad-CAM as possible, we minimize the influence caused by information of pixels. We thus prove the following proposition.

**Proposition 1.** Let  $I_N^*$  be the reference set such that  $\forall 0 < N \leq d$ ,  $|I_N^*| = N$ , and  $I_N^* \subset I_{N+1}^*$ .  $\forall 0 < n \leq N$ , let  $I_{n,N} \subset I_N^*$  be a random subset of the cardinality  $n$ . The



Figure 5: Illustrations of masked images in Pixel Flipping of VGG-16. For each sample,  $n = 9k$  pixels are masked. The first row is random masking. The second row is random masking with Grad-CAM reference (i.e., the masked 9k pixels are randomly chosen from the  $N=12k$  pixels based on Grad-CAM). The bottom row is Grad-CAM masking.

expected Dice similarity (Dice, 1945) between  $I_{n,N}$  and  $I_n^*$  is decreasing w.r.t.  $N$  in inverse proportion.

Please refer to Appendix F for the proof. Let  $d \geq N \geq n$  denote the number of all pixels, referenced pixels, and masked pixels, respectively. Based on this proposition, if  $N = d$ , the referenced masking degenerates to totally random masking. If  $N = n$ , it degenerates to Grad-CAM masking. So we need to balance  $N$  between  $n$  and  $d$ . We let  $N$  vary from 15k to 30k, and  $n$  vary from 0 to  $N$ . It should be noticed that this is an approximation, because there are three variables, the masked area (object or background), the pixel isolation, and the number of masked pixels, and it is impossible to alter only one of them without changing others. We fix the number of masked pixels and make compromises between the area and the isolation. The results are shown in Figure 6.

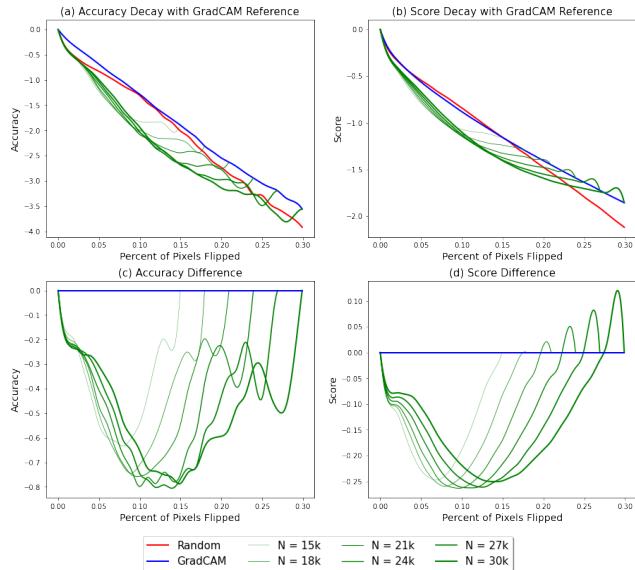


Figure 6: Pixel Flipping results with Grad-CAM reference. Blue, red, green curves represent Grad-CAM, Random, and Random with Grad-CAM reference, respectively. Different widths of green curves represent different numbers of referenced pixels  $N$ . (a)(c) are the results of accuracy and (b)(d) are the results of scores. (c)(d) are the differences between the referenced masking and the Grad-CAM (reference) masking. All results are normalized to  $[0, 1]$  and then taken the logarithm for better illustration.

As the combinations of random masking and Grad-CAM masking, all referenced masking results outperform these two ingredients. This corresponds to the fact that isolated masking methods such as Gradient outperform unisolated masking methods like Grad-CAM. Also, from Figure 6(c)(d), the lowest point of each referenced masking curve locates near the midpoint of the corresponding  $N$ . This suggests that both isolation and the information of pixels contribute greatly to the performance decay. As  $n \rightarrow N$ , the effect of isolation diminishes, so the referenced masking converges to the Grad-CAM reference.

#### 6.2.4 BASELINE FLAWS

When a pixel is masked, numerically, there must be some other value to replace it, which is often referred to as the baseline (Sundararajan et al., 2017). The baseline represents the null feature in the input that provides no information. It is generally set to simple intuitive values like zeros, means, random noises, etc. Recently, there are other complex baselines such as Gaussian blur (Sturmels et al., 2020), or even inpainting algorithms (Samek et al., 2021). Unfortunately, there is no baseline theoretically justified as “truly non-informative”. All proposed baselines are approximations and make compromises. Seeking the best baseline

is beyond the scope of our work. Therefore, as we want to introduce as little exogenous information as possible, we use simple baselines such as zeros or random noises in experiments.

## 7 Application of Faithfulness and Plausibility

Based on the importance and the chaos of plausibility and faithfulness, we present several applications of them with the goal of improving the interpretability research.

**Stop using plausibility solely as the measure.** It has been argued that visualizations should not be used solely as the measure for explanations (Leavitt and Morcos, 2020). We emphasize that not only visualizations, but also their superset, the plausibility, should not be used solely. Compared with visualizations, it includes quantitative plausibility-based metrics, such as Pointing Game, BAM, etc. Even though plausibility has been extensively applied in XAI, an implausible explanation is not necessarily less desired than a more plausible one. They should not be intentionally avoided. In fact, it is those implausible explanations that are really useful. Because they may be actually revealing the implausible mechanism of black boxes.

**Faithfulness should be handled carefully.** For DNNs, the ground truth of faithfulness of explanations still remains an open question (Jacovi and Goldberg, 2020). Currently, with existing techniques, it is impossible to decide directly whether the explanation is faithful or not. We suggest that researchers should not claim faithfulness unless this issue has been solved. This does not mean faithfulness should be neglected. On the contrary, it should be carefully considered in developing and evaluating explanation methods.

**Metrics should introduce as little bias as possible.** With strong assumptions and exogenous information introduced, a metric can be completely biased as shown in Section 6. Such evaluation will only lead to counter-productive results.

**The Goals of explanation metrics and methods should be clear and compatible.** Different explanations methods can serve different goals. We suggest that it is unreasonable to use universal metrics to evaluate these explanations altogether. Therefore, the goal of a metric needs to be clear and consistent with the explanation. Simultaneously, the goals of explanations should also be clear.

**It is the usage instead of the superiority of explanations that matters.** Given the insufficient study on the metrics, we encourage researchers not to be stuck in this chaos. Instead of trying to demonstrate the superiority of their explanation methods to others, the usage of explanation methods should draw more attention. An explanation is really desired if it

can help in practical applications, such as debugging, or avoiding the overfitting phenomenon.

## 8 Conclusion

From the perspective of evaluating explanations, we study existing explanation metrics of two categories, the alignment metrics, and the performance metrics. Based on this, we analyze fatal flaws in nowadays' XAI research from many aspects. We then demonstrate that such flaws are caused by the chaos of explanation metrics. We also present several experiments to show the importance of proper metrics. Finally, we propose suggestions on both developing and evaluating explanations for the XAI research. We encourage researchers to take these criteria into careful consideration.

We admit that building perfect explanations and developing perfect metrics for explanations are two complementary tasks. Neither of them is fulfilled yet. The goal of this paper is to provide a unified study that clearly analyzes them, and that can be potentially useful for exploring such perfection.

## References

- Adebayo, J., Gilmer, J., Muelly, M., Goodfellow, I., Hardt, M., and Kim, B. (2018). Sanity checks for saliency maps. *arXiv preprint arXiv:1810.03292*.
- Adebayo, J., Muelly, M., Liccardi, I., and Kim, B. (2020). Debugging tests for model explanations. *arXiv preprint arXiv:2011.05429*.
- Agarwal, R., Frosst, N., Zhang, X., Caruana, R., and Hinton, G. E. (2020). Neural additive models: Interpretable machine learning with neural nets. *arXiv preprint arXiv:2004.13912*.
- Alvarez-Melis, D. and Jaakkola, T. S. (2018). Towards robust interpretability with self-explaining neural networks. *arXiv preprint arXiv:1806.07538*.
- Arras, L., Horn, F., Montavon, G., Müller, K.-R., and Samek, W. (2016). Explaining predictions of non-linear classifiers in nlp. *arXiv preprint arXiv:1606.07298*.
- Bach, S., Binder, A., Montavon, G., Klauschen, F., Müller, K.-R., and Samek, W. (2015). On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PloS one*, 10(7):e0130140.
- Carter, B., Jain, S., Mueller, J., and Gifford, D. (2020). Overinterpretation reveals image classification model pathologies. *arXiv preprint arXiv:2003.08907*.
- Chen, C., Li, O., Tao, C., Barnett, A. J., Su, J., and Rudin, C. (2018a). This looks like that: deep learning for interpretable image recognition. *arXiv preprint arXiv:1806.10574*.
- Chen, C., Yuan, J., Lu, Y., Liu, Y., Su, H., Yuan, S., and Liu, S. (2020). Oodanalyzer: Interactive analysis of out-of-distribution samples. *IEEE Transactions on Visualization and Computer Graphics*, 27(7):3335–3349.
- Chen, J., Song, L., Wainwright, M. J., and Jordan, M. I. (2018b). L-shapley and c-shapley: Efficient model interpretation for structured data. *arXiv preprint arXiv:1808.02610*.
- Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255. Ieee.
- Dice, L. R. (1945). Measures of the amount of ecologic association between species. *Ecology*, 26(3):297–302.
- Everingham, M., Van Gool, L., Williams, C. K., Winn, J., and Zisserman, A. (2010). The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88(2):303–338.
- Fong, R., Patrick, M., and Vedaldi, A. (2019). Understanding deep networks via extremal perturbations and smooth masks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2950–2958.
- Geirhos, R., Rubisch, P., Michaelis, C., Bethge, M., Wichmann, F. A., and Brendel, W. (2018). Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. *arXiv preprint arXiv:1811.12231*.
- Ghorbani, A., Wexler, J., Zou, J., and Kim, B. (2019). Towards automatic concept-based explanations. *arXiv preprint arXiv:1902.03129*.
- Hooker, S., Erhan, D., Kindermans, P.-J., and Kim, B. (2018). A benchmark for interpretability methods in deep neural networks. *arXiv preprint arXiv:1806.10758*.
- Jacovi, A. and Goldberg, Y. (2020). Towards faithfully interpretable nlp systems: How should we define and evaluate faithfulness? *arXiv preprint arXiv:2004.03685*.
- Kim, B., Wattenberg, M., Gilmer, J., Cai, C., Wexler, J., Viegas, F., et al. (2018). Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In *Proceedings of the 35th International Conference on Machine Learning*, pages 2668–2677. PMLR.

- Koh, P. W. and Liang, P. (2017). Understanding black-box predictions via influence functions. In *Proceedings of the 34th International Conference on Machine Learning*, pages 1885–1894. PMLR.
- Koh, P. W., Nguyen, T., Tang, Y. S., Mussmann, S., Pierson, E., Kim, B., and Liang, P. (2020). Concept bottleneck models. In *Proceedings of the 37th International Conference on Machine Learning*, pages 5338–5348. PMLR.
- Krizhevsky, A. (2014). One weird trick for parallelizing convolutional neural networks. *arXiv preprint arXiv:1404.5997*.
- Leavitt, M. L. and Morcos, A. (2020). Towards falsifiable interpretability research. *arXiv preprint arXiv:2010.12016*.
- Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In *Proceedings of the European Conference on Computer Vision*, pages 740–755. Springer.
- Lin, Z. Q., Shafiee, M. J., Bochkarev, S., Jules, M. S., Wang, X. Y., and Wong, A. (2019). Do explanations reflect decisions? a machine-centric strategy to quantify the performance of explainability algorithms. *arXiv preprint arXiv:1910.07387*.
- Lundberg, S. M. and Lee, S.-I. (2017). A unified approach to interpreting model predictions. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 4768–4777.
- Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. (2017). Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*.
- Miller, T. (2019). Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence*, 267:1–38.
- Parekh, J., Mozharovskyi, P., and d’Alché Buc, F. (2020). A framework to learn with interpretation. *arXiv preprint arXiv:2010.09345*.
- Petsiuk, V., Das, A., and Saenko, K. (2018). Rise: Randomized input sampling for explanation of black-box models. *arXiv preprint arXiv:1806.07421*.
- Poerner, N., Roth, B., and Schütze, H. (2018). Evaluating neural network explanation methods using hybrid documents and morphological agreement. *arXiv preprint arXiv:1801.06422*.
- Preece, A., Harborne, D., Braines, D., Tomsett, R., and Chakraborty, S. (2018). Stakeholders in explainable ai. *arXiv preprint arXiv:1810.00184*.
- Ribeiro, M. T., Singh, S., and Guestrin, C. (2016). ” why should i trust you?” explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1135–1144.
- Rudin, C. (2019). Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1(5):206–215.
- Samek, W., Binder, A., Montavon, G., Lapuschkin, S., and Müller, K.-R. (2016). Evaluating the visualization of what a deep neural network has learned. *IEEE Transactions on Neural Networks and Learning Systems*, 28(11):2660–2673.
- Samek, W., Montavon, G., Lapuschkin, S., Anders, C. J., and Müller, K.-R. (2021). Explaining deep neural networks and beyond: A review of methods and applications. *Proceedings of the IEEE*, 109(3):247–278.
- Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., and Batra, D. (2017). Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 618–626.
- Shah, H., Jain, P., and Netrapalli, P. (2021). Do input gradients highlight discriminative features? *arXiv preprint arXiv:2102.12781*.
- Shrikumar, A., Greenside, P., and Kundaje, A. (2017). Learning important features through propagating activation differences. In *Proceedings of the 34th International Conference on Machine Learning*, pages 3145–3153. PMLR.
- Simonyan, K., Vedaldi, A., and Zisserman, A. (2013). Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*.
- Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.
- Springenberg, J. T., Dosovitskiy, A., Brox, T., and Riedmiller, M. (2014). Striving for simplicity: The all convolutional net. *arXiv preprint arXiv:1412.6806*.
- Srinivas, S. and Fleuret, F. (2020). Rethinking the role of gradient-based attribution methods for model interpretability. *arXiv preprint arXiv:2006.09128*.
- Sturmels, P., Lundberg, S., and Lee, S.-I. (2020). Visualizing the impact of feature attribution baselines. *Distill*, 5(1):e22.

Sundararajan, M., Taly, A., and Yan, Q. (2017). Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning*, pages 3319–3328. PMLR.

Tan, M., Chen, B., Pang, R., Vasudevan, V., Sandler, M., Howard, A., and Le, Q. V. (2019). Mnasnet: Platform-aware neural architecture search for mobile. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2820–2828.

Tomsett, R., Harborne, D., Chakraborty, S., Gurram, P., and Preece, A. (2020). Sanity checks for saliency metrics. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 6021–6029.

Wang, Y. and Wang, X. (2021). Self-interpretable model with transformation equivariant interpretation. *Advances in Neural Information Processing Systems*, 34.

Yang, M. and Kim, B. (2019). Benchmarking attribution methods with relative feature importance. *arXiv preprint arXiv:1907.09701*.

Yeh, C.-K., Hsieh, C.-Y., Suggala, A., Inouye, D. I., and Ravikumar, P. K. (2019). On the (in) fidelity and sensitivity of explanations. *Advances in Neural Information Processing Systems*, 32:10967–10978.

Zeiler, M. D. and Fergus, R. (2014). Visualizing and understanding convolutional networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 818–833. Springer.

Zhang, J., Bargal, S. A., Lin, Z., Brandt, J., Shen, X., and Sclaroff, S. (2018). Top-down neural attention by excitation backprop. *International Journal of Computer Vision*, 126(10):1084–1102.

Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., and Torralba, A. (2016). Learning deep features for discriminative localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2921–2929.

Zhou, B., Sun, Y., Bau, D., and Torralba, A. (2018). Interpretable basis decomposition for visual explanation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 119–134.

Zhou, Y., Booth, S., Ribeiro, M. T., and Shah, J. (2021). Do feature attribution methods correctly attribute features? *arXiv preprint arXiv:2104.14403*.

## A Related Work Supplement

Based on the stage where the explanations are generated, attribution methods can further be divided as post-hoc methods and self-interpretable methods. Post-hoc methods are already included in Section 2. They are developed to explain pre-trained models (usually DNNs). However, it is also argued that post-hoc methods are not reliable and hence not trusted by humans, and that self-interpretable models are more desired (Rudin, 2019). Self-interpretability is acknowledged as the property that the explanations and the predictions are generated at the same stage, and that the explanations are directly involved in the predictions. Within this genre, white-box models such as sparse linear models, decision tree, etc. are usually treated as axiomatically self-interpretable. However, their expressiveness is limited by their low complexity. Therefore, there are also deep models arguably claiming self-interpretability by regularizing the models themselves (Alvarez-Melis and Jaakkola, 2018; Chen et al., 2018a; Agarwal et al., 2020; Wang and Wang, 2021), etc. But these models are still limited by universality, complexity, expressiveness, etc.

## B PGD Enhancement

### B.1 Implementation Details

The standard PGD attack (Madry et al., 2017) is defined as follows

$$\mathbf{x}^{t+1} = \Pi_{\mathbf{x}+S}(\mathbf{x}^t + \alpha \operatorname{sgn}(\nabla_{\mathbf{x}} L(\theta, \mathbf{x}, y))),$$

where  $\alpha$  is the step size, and  $S$  is the set of allowed perturbations.  $S$  is defined as a box area  $[-\epsilon, \epsilon]^d$ . The PGD enhancement is then defined as

$$\mathbf{x}^{t+1} = \Pi_{\mathbf{x}+S}(\mathbf{x}^t - \alpha \operatorname{sgn}(\nabla_{\mathbf{x}} L(\theta, \mathbf{x}, y))).$$

All PGD enhancement examples shown in this paper are generated under the following settings:  $\epsilon = 0.3$ ,  $\alpha = 0.03$ , and the number of steps is 30.

### B.2 PGD Enhancement for Other Post-Hoc Explanations

Here we present illustration of PGD-enhancement results of different post-hoc explanation methods. Apart from Grad-CAM (Selvaraju et al., 2017) shown in Section 4, we also present Excitation Backpropagation (Zhang et al., 2018) in Figure 1, Linear Approximation in Figure 2, Guided Backpropagation (Springenberg et al., 2014) in Figure 3, Gradient (Simonyan et al., 2013) in Figure 4, and DeConvNet (Zeiler and Fergus, 2014) in Figure 5. It can be found that The explanations of enhanced images and the untrained model (middle) are more similar to the explanations of raw images and the untrained model (left), rather than those of

raw images and the trained model (right). However, the predicted classification results of the middle columns are the same as the right columns, which are correct. The predicted classification results of the left columns are wrong. This phenomenon is because while the predictions of the middle columns and the right columns are same, their true mechanism (for DNNs, is the collection of weights) are completely different. And this difference in *faithfulness* is captured by most explanation methods. However, with a plausibility-based metric, these explanation methods (for the middle columns) will be measured unfairly.

### B.3 Pixel Isolation

Comparing the explanations of Guided Backpropagation (Figure 3), Gradient (Figure 4), DeConvNet (Figure 5) with others, we can find that the explanations of these three methods have the property *pixel isolation*. That is, the pixels with adjacent attribution values are always isolated. Hence their highlighted areas in heatmaps are many isolated pixels, while the highlighted areas of Grad-CAM (Section 4), Excitation Backpropagation (Figure 1), and Linear Approximation (Figure 2) are consecutive pixel patches. This difference is one of the most exogenous information introduced when applying Pixel Flipping or other metrics that requires modifying the input images. More details about the studies on this difference are presented in Section 6.

## C Pointing Game

Given an image instance  $\mathbf{x}$ , a classifier  $F$ , and a targeted class  $t$ , an attribution method usually returns a attribution map as the explanation for the targeted class. From the quantitative attribution map, there's a pixel that can be recognized as the “most important”. Pointing Game then calculates the ratio that the most important pixel falls into the “object areas”. Such areas are usually in the form of pre-annotated bounding boxes or silhouettes of the objects. Formally, suppose there are  $N$  instances, and the 2-D coordinate of the most important pixel of  $\mathbf{x}_i$  is  $(u_i, v_i)$ , and the annotated object area of  $\mathbf{x}_i$  for target  $t_i$  is  $\Omega_i$ . The explanation method scores a hit if  $\max_{(u,v) \in \Omega_i} \|(u_i, v_i) - (u, v)\|_2 \leq \tau$  where  $\tau$  is a predefined tolerance, usually applied to mitigate the error introduced by upsampling in some attribution methods. Then the Pointing Game score is defined by the ratio of number of total hits to the number of all samples, as

$$r = \frac{\# \text{ of hits}}{\# \text{ of samples}}.$$

## D Annotated MNIST

Since the background pixels of MNSIT data all have value 0, we automatically drawing a bounding box for each image

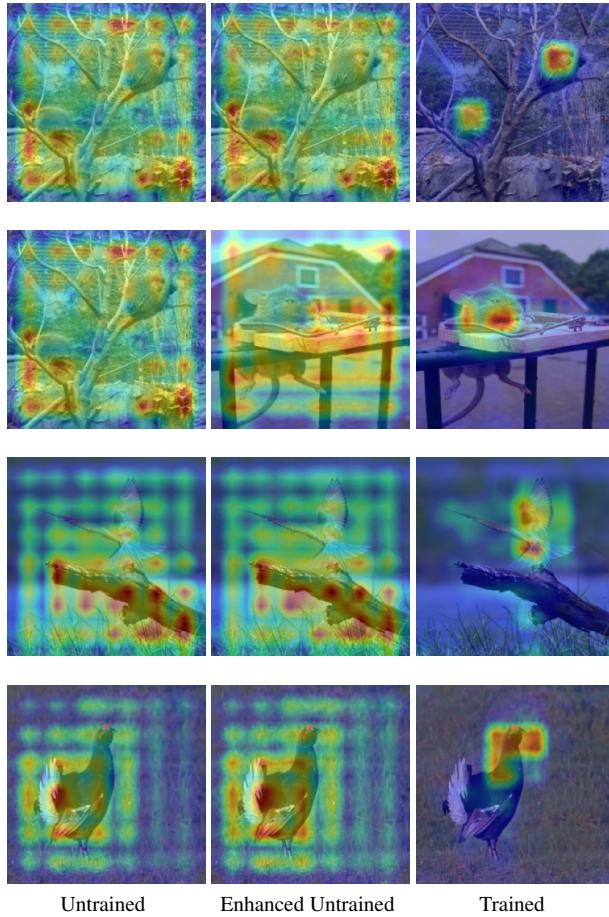


Figure 1: Excitation Backpropagation explanations of VGG-16 models. The results are from the untrained model with raw data as input (top row), the untrained model with PGD-enhanced data as input (middle row), and the trained model with raw data as input (bottom row). All samples of the middle and the bottom rows are correctly classified, and all samples from the top row are falsely classified.

of MNSIT dataset, such that all pixels with positive values are included in the bounding box. The edges of bounding boxes are drawn exactly next to the marginal pixels of the corresponding image margins. Some examples are shown in Figure 6. In Pointing Game, an explanation method scores a hit if the pixel with the highest attribution value falls strictly within (not on the edges) the corresponding bounding box. Also, since in the experiments we apply linear models, which do not require upsampling, the tolerance  $\tau$  is set to zero.

## E Accuracy as Indicator in Pixel Flipping

Suppose we have a linear classifier, where we treat the weight  $\mathbf{w}_i \odot \mathbf{x}$  as the explanations. Suppose we have

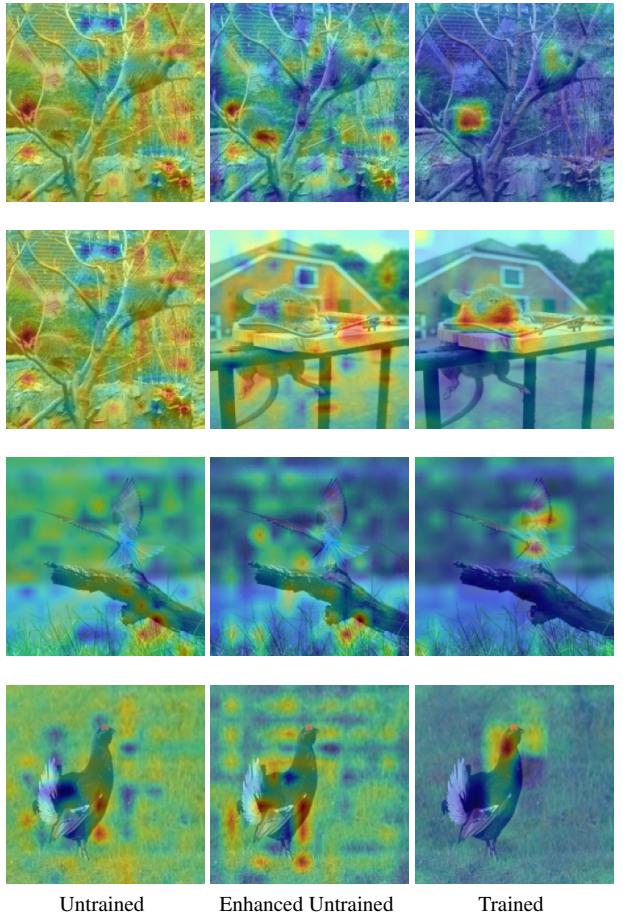


Figure 2: Linear Approximation explanations of VGG-16 models. The results are from the untrained model with raw data as input (top row), the untrained model with PGD-enhanced data as input (middle row), and the trained model with raw data as input (bottom row). All samples of the middle and the bottom rows are correctly classified, and all samples from the top row are falsely classified.

$\mathbf{x} = [x_1, \dots, x_d] \in \mathbb{R}^d$ , and  $W = [\mathbf{w}_1, \dots, \mathbf{w}_c] \in \mathbb{R}^{d \times c}$ , then the prediction is  $\mathbf{y} = W^\top \mathbf{x} \in \mathbb{R}^c$ . Here we omit the softmax activation since it does not change the relative relations. The classification result is then decided by  $\hat{\mathbf{y}} = \arg \max_{1 \leq i \leq c} y_i$ . Suppose  $t \in [c]$  to be the target. Since most attribution methods only build the attribution map based on a specific class, (That is, there are  $c$  attribution maps, each of which is corresponding to a specific class.) Pixel Flipping method only flips the values of  $x_i$  according to the corresponding value  $w_t^i$ . Without loss of generality, we set  $w_t^1 x_1 \geq w_t^2 x_2 \geq \dots \geq w_t^d x_d$  so that we can process the masking by this order. Then by the criterion of Pixel Flipping, we mask  $x_1, x_2, \dots, x_n$  one by one. Suppose the instance  $\mathbf{x}$  is classified correctly, then without loss of gener-

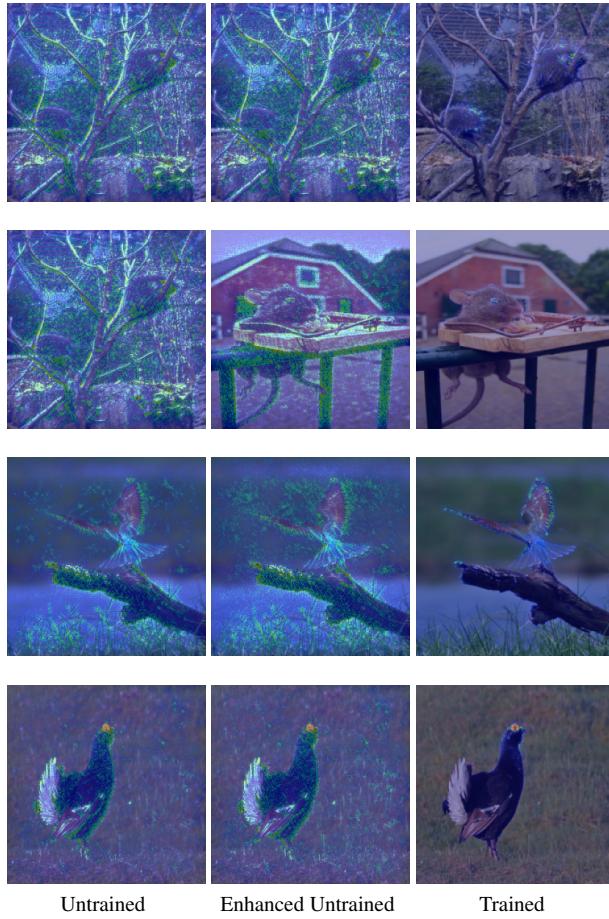


Figure 3: Guided Backpropagation explanations of VGG-16 models. The results are from the untrained model with raw data as input (top row), the untrained model with PGD-enhanced data as input (middle row), and the trained model with raw data as input (bottom row). All samples of the middle and the bottom rows are correctly classified, and all samples from the top row are falsely classified.

ability, we can also set  $\mathbf{w}_1^\top \mathbf{x} \geq \mathbf{w}_2^\top \mathbf{x} \geq \dots \geq \mathbf{w}_c^\top \mathbf{x}$ . Based on the setting specified above, Pixel Flipping can be easily disproved. The original prediction process is

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_c \end{bmatrix} = \begin{bmatrix} w_1^1 & w_1^2 & \dots & w_1^d \\ w_2^1 & w_2^2 & \dots & w_2^d \\ \vdots & \vdots & \ddots & \vdots \\ w_c^1 & w_c^2 & \dots & w_c^d \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_d \end{bmatrix}$$

After masking the most relevant feature, we have

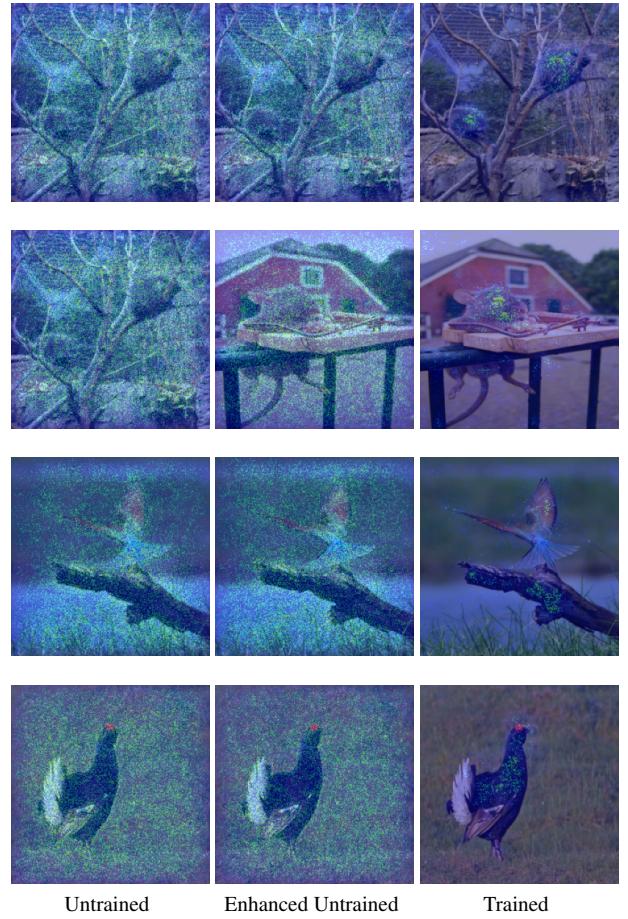


Figure 4: Gradient explanations of VGG-16 models. The results are from the untrained model with raw data as input (top row), the untrained model with PGD-enhanced data as input (middle row), and the trained model with raw data as input (bottom row). All samples of the middle and the bottom rows are correctly classified, and all samples from the top row are falsely classified.

$$\begin{bmatrix} y_1 - w_1^1 x_1 \\ y_2 - w_2^1 x_1 \\ \vdots \\ y_c - w_c^1 x_1 \end{bmatrix} = \begin{bmatrix} w_1^1 & w_1^2 & \dots & w_1^n \\ w_2^1 & w_2^2 & \dots & w_2^n \\ \vdots & \vdots & \ddots & \vdots \\ w_c^1 & w_c^2 & \dots & w_c^n \end{bmatrix} \begin{bmatrix} 0 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

Trivially, although  $y_1 = \mathbf{w}_1^\top \mathbf{x} \geq \mathbf{w}_2^\top \mathbf{x} = y_2$ , and  $w_1^1 x_1 > w_i^1 x_i$  for  $\forall i > 1$ , it is not guaranteed that masking  $x_1$  can result in the best performance decay. This is because the difference between the distributions of different weight vector  $\mathbf{w}_i$ . This can be verified by a very simple example. Let

$$\mathbf{x} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, W = [\mathbf{w}_1 \quad \mathbf{w}_2 \quad \mathbf{w}_3] = \begin{bmatrix} 2 & \sqrt{6} & \sqrt{5} \\ 1 & 0 & 0 \end{bmatrix}$$



Figure 5: DeConvNet explanations of VGG-16 models. The results are from the untrained model with raw data as input (top row), the untrained model with PGD-enhanced data as input (middle row), and the trained model with raw data as input (bottom row). All samples of the middle and the bottom rows are correctly classified, and all samples from the top row are falsely classified.

We then have the preconditions  $y_1 = 3 > \sqrt{6} = y_2 > \sqrt{5} = y_3, w_1^1 x_1 = 2 > 1 = w_1^2 x_2$ . However, masking  $x_1$  will result in

$$\begin{aligned} y'_1 &= y_1 - w_1^1 x_1 = 1 \\ y'_2 &= y_2 - w_2^1 x_1 = 0 \\ y'_3 &= y_3 - w_3^1 x_1 = 0 \end{aligned}$$

The classification result will not be changed. If we instead mask the feature  $x_2$ , which has lower explanation value, we

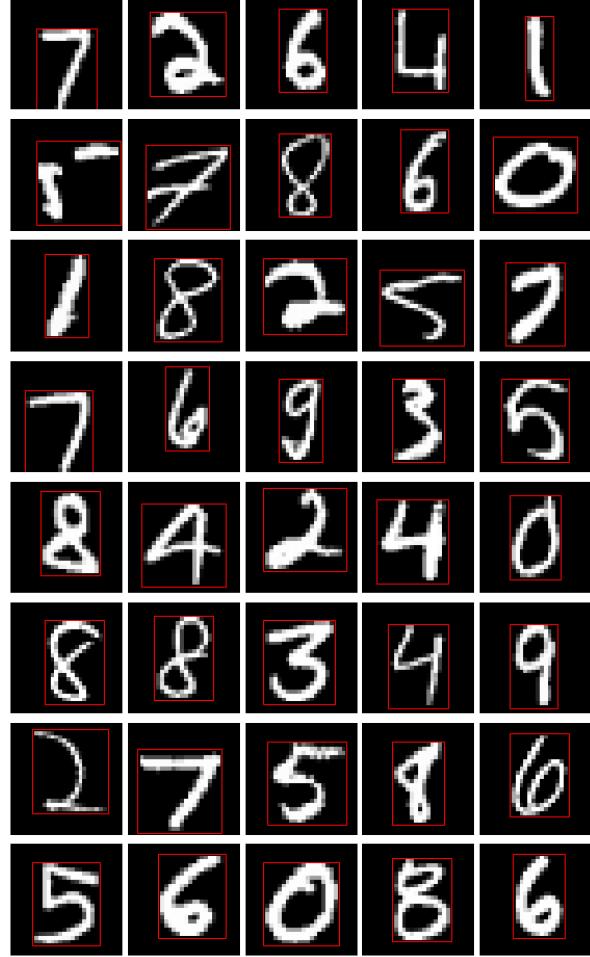


Figure 6: MNIST Data Examples with Bounding Box annotations.

then have

$$\begin{aligned} y'_1 &= y_1 - w_1^2 x_2 = 2 \\ y'_2 &= y_2 - w_2^2 x_2 = \sqrt{6} \\ y'_3 &= y_3 - w_3^2 x_2 = \sqrt{5} \end{aligned}$$

which changes the accuracy. This example shows that accuracy should not be used as the indicator, even for the linear model.

## F Proof of Proposition 1

Suppose  $\mathbf{x} = (x_1, \dots, x_d)$  is the input data. Without loss of generality, suppose they are ordered monotonically decreasing by the order of attribution values from some explanations.  $I_n^* = [n]$ , and  $I_{n,N} \subset [N]$  are two index subsets, where  $I_{n,N}$  is randomly sampled such that  $|I_{n,N}| = n$ . Denote the Dice similarity as  $s(A, B) = \frac{2|A \cap B|}{|A| + |B|}$ , then the expected distance

between  $I_{n,N}$  and  $I_n^*$  is defined as

$$\begin{aligned}
D(N) &= \mathbb{E}_{I_{n,N} \in I_N^*} [s(I_n^*, I_{n,N})] \\
&= \sum_{k=\max\{0,2n-N\}}^n \frac{\binom{n}{k} \binom{N-n}{n-k} k}{\binom{N}{n}} \frac{k}{n} \\
&= \frac{1}{n \binom{N}{n}} \sum_{k=\max\{0,2n-N\}}^n k \binom{n}{k} \binom{N-n}{n-k} \\
&= \frac{1}{n \binom{N}{n}} \cdot n \binom{N-1}{n-1}
\end{aligned}$$

which is a inverse proportional function. The above derivation is based on the following reasoning. According to the binomial theorem, on the one hand

$$n(1+x)^{N-1} = \sum_{k=1}^N n \binom{N-1}{k-1} x^{k-1}$$

on the other hand,

$$\begin{aligned}
n(1+x)^{N-1} &= n(1+x)^{n-1}(1+x)^{N-n} \\
&= n \left( \sum_{i=0}^{n-1} \binom{n-1}{i} x^i \right) \left( \sum_{j=0}^{N-n} \binom{N-n}{j} x^j \right) \\
&\stackrel{k=i+j}{=} \sum_{i=1}^n \sum_{k=i}^{N-n+i} i \binom{n}{i} \binom{N-n}{k-i} x^{k-1} \\
&= \sum_{k=1}^{N-n} \sum_{i=\max\{1,k-N+n\}}^{\min\{k,n\}} i \binom{n}{i} \binom{N-n}{k-i} x^{k-1}
\end{aligned}$$

Comparing the coefficients of  $x^{n-1}$ , we have

$$\sum_{i=\max\{0,2n-N\}}^n i \binom{n}{i} \binom{N-n}{n-i} = n \binom{N-1}{n-1}.$$

Therefore,  $D(N) = \frac{1}{n \binom{N}{n}} \cdot n \binom{N-1}{n-1} = \frac{n}{N}$ .  $\square$