# Interpreting Interpretations: Organizing Attribution Methods by Criteria

Zifan Wang, Piotr Mardziel, Anupam Datta, Matt Fredrikson
Carnegie Mellon Univeristy
Moffett Field, CA, 94089
zifanw@andrew.cmu.edu

## Abstract

*Motivated by distinct, though related, criteria, a growing number of attribution methods have been developed to interprete deep learning. While each relies on the interpretability of the concept of "importance" and our ability to visualize patterns, explanations produced by the methods often differ. In this work we expand the foundations of human-understandable concepts with which attributions can be interpreted beyond "importance" and its visualization; we incorporate the logical concepts of necessity and sufficiency, and the concept of proportionality. We define metrics to represent these concepts as quantitative aspects of an attribution. We evaluate our measures on a collection of methods explaining convolutional neural networks (CNN) for image classification. We conclude that some attribution methods are more appropriate for interpretation in terms of necessity while others are in terms of sufficiency, while no method is always the most appropriate in terms of both.*

## 1. Introduction

Among approaches for interpreting opaque models are input attribution which assign to each input a level of contribution to its output. When visualized alongside inputs, an attribution gives a human interpreter a notion of what about images is important to the prediction (see, for example, Figure 2). Being explanations of highly complex systems intended for highly complex humans, attributions have been varied in their approaches and sometimes produce distinct explanations even for the same instances.

Nevertheless, save for the earliest approaches, attribution methods distinguish themselves with one or more desirable criteria. Scaling criteria such as *completeness* [? ], *sensitivity-n* [? ], *linear-agreement* [? ? ] calibrate attribution to the change in output as compared to change in input when evaluated on some baseline. Given access to different attribution methods, which one is the optimal choice for what purpose remains an unexplored area. Visual comparisons, though intuitive and straightforward, remain less objective since 1) human themselves do not agree often and 2) attribution maps generated by different methods may vary or even cause contradictory interpretations (see Fig 1 for example).

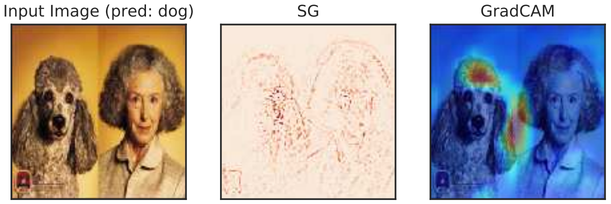

Figure 1. Differences in explanations for a neural network prediction. Left: the input with predicted class and groundtruth `dog`. Middle: SmoothGrad [? ]. Pixels with deeper color have higher attribution. Right: GradCAM[? ]. Regions with more heat localize the more relevant spatial locations. Questions: *Is the model using the lady for predicting dog? Which explanation is accurate?*

While evaluation criteria endow attributions with some limited semantics, the variations in design goals, evaluation metrics, and the underlying methods resulted in attributions failing at their primary goal: aiding in model interpretation. This work alleviates these problems and makes the following contributions.

- We decompose and organize existing attribution methods' goals along two complementary properties: ordering and proportionality. While ordering requires that an attribution should order input features according to some notion of importance, proportionality stipulates also a quantitative relationship between a model's outputs and the corresponding attributions in that particular ordering.

- We describe how all existing methods are motivated by an attribution ordering corresponding roughly to the logical notion of necessity which leads to a corresponding sufficiency ordering not yet fully discussed in literature.

- We show that while some attribution methods show great performance in necessity while others show more about sufficiency but no evaluated method in this paper can be a winner on the necessity and sufficiency at the same time.

- We further demonstrate how to interpret different attribution maps to gain more insights about the decision making process in deep models.

## 2. Background

Attributions are a simple form of model explanations that have found significant application to Convolutional Neural Networks (CNNs) with their ease of visualization alongside model inputs (i.e. images). We summarize the various approaches in Section 2.1 and the criteria with which they are evaluated and/or motivated in Section 2.3.

### 2.1. Attribution Methods

The concept of Attribution is well-defined in [? ] but it excludes any method without an baseline (reference) input. We consider a relaxed version. Consider a classification model $\mathbf{y} = M(\mathbf{x})$ that takes an input vector $\mathbf{x}$ and outputs a score vector $\mathbf{y} = [y_0, \cdots, y_i, \cdots, y_{n-1}]^\top$, where $y_i$ is the score of predicting $\mathbf{x}$ as class $i$ and there are $n$ classes in total. Given a pre-selected class $c$, an attribution method attempts to explain $y_c$ by computing a score for each feature $x_i$ as its contribution toward $y_c$. Even though each feature in $\mathbf{x}$ may receive different attribution scores given different choice of attribution methods, features with positive attribution scores are universally explained as important part in $\mathbf{x}$, while the negative scores indicate the presence of these features decline the confidence for predicting $y_c$.

Previous work has made great progress in developing gradient-based attribution methods to highlight important features in the input image for explaining model's prediction. The primary question to answer is whether should we consider *grad* or *grad* × *input* as attributions [? ? ? ? ]. As ? ] argues *grad* is *local attribution* that only accounts for how tiny change around the input will influence the output of the network but *grad* × *input* is the *global attribution* that accounts for the marginal effect of a feature towards output. We use *grad* × *input* as the attribution to be discussed in this paper. We briefly introduce methods to be evaluated in this paper and examples are provided in Fig 2.

**Saliency Map** (SM) [? ? ] uses the gradient of the class of interests with respect to the input to interpret the prediction result of CNNs. **Guided Backpropagation** (GB) [? ] modifies the backpropagation of ReLU [? ] so that only the positive gradients will be passed into the previous layers. **GradCAM** [? ] builds on the Class Activation Map (CAM) [? ] targeting CNNs. Although its variations [? ? ] show sharper visualizations, their fundamental concepts remain unchanged. We consider only GradCAM in this paper. **Layer-wise Relevance Propagation** (LRP) [? ], **DeepLift** [? ] modifies the local gradient and rules of backpropagations. Another method sharing similar motivation in design with DeepLift is **Integrated Gradient** (IG) [? ]. IG computes attribution by integrating the gradient over a path from a pre-defined baseline to the input. **SmoothGrad** (SG) [? ] attempts to denoise the result of Saliency Map by adding Gaussian noise to the input and provides visually sharper results.

Other methods like Deep Taylor Decompostion [? ] related with LRP, Occluding [? ] and Influence Directed Explanations [? ] are not evaluated in this paper but will be a proper future work to discuss.

### 2.2. Assumptions

We restrict ourselves with two assumptions with regards to models and attribution methods analyzed.

**Non-linearity** We focus on evaluating the performance of attribution methods on non-linear model, *e.g.* neural networks, as SM, IG, SG, LRP, and DeepLIFT are equivalent for linear models (see proofs in Appendix I) while GradCAM only works for convolutional layers. Linear models are therefore not expected to distinguish most attribution methods.

**Feature Interaction** Features may or may not influence the decision individually. In this paper, we focus on attribution methods that are not directly suited to reasoning about feature interaction: their attribution maps represent per-pixel importance, and do not indicate relationships between pixels. We are interested in evaluating the feature interactions in the future work.

### 2.3. Evaluation Criteria

Evaluation criteria measure the performance of attribution methods towards some desirable characteristic and are typically employed to justify the use of novel attribution methods. We begin with discussing two assumptions about evaluating the attribution methods.

The most common evaluations are based on pixel-level interventions or perturbations. These quantify the correlation between the perturbed pixels' attribution scores and the output change [? ? ? ? ? ? ? ]. For perturbations that intend to remove or ablate a pixel (typically by setting it to some baseline or to noise), the desired behavior for an optimal attribution method is to have perturbations on the highly attributed pixels drop the class score more significantly than on the pixels with lower attribution.

Quantification of the behavior described by ? ] with *Area Over Perturbation Curve (AOPC)* measures the area between two curves: the model's output score against the number of perturbed pixels in the input image and the horizontal line of the score at the model's original output. Two

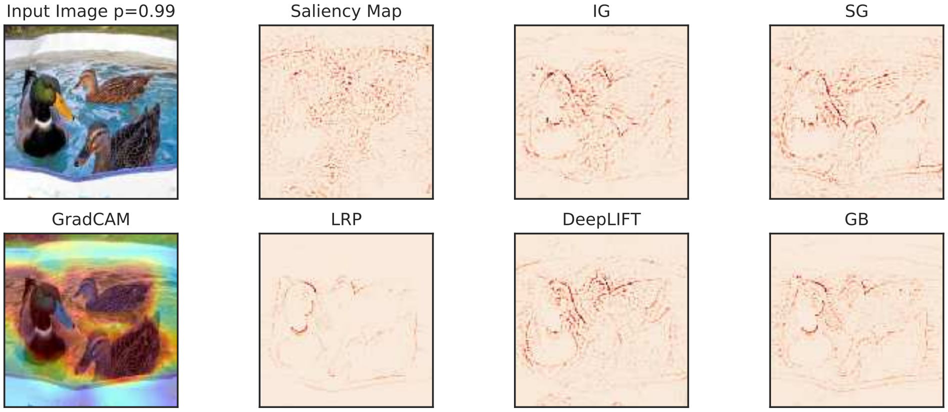

Figure 2. Visualizations of different attribution methods using VGG16 model [? ]. (a) is the input image with an output confidence 0.99 for the true label `duck` Different attribution methods are discussed in Section 2.1. *grad* × *input* is applied to Saliency Map, IG, SG and GB. We only use heatmap for GradCAM to align with the choice of visualization in ? ].

similar measurement are *Area Under Curve (AUC)* [? ? ] and *MOst Relevant features First (MoRF)* [? ] that measure the area under the perturbation curve instead. AOPC and AUC (we use AUC to represent both AUC and MoRF) measurement are equivalent and both are orignally used to endorse LRP. For reasons which will become clear in the next section, we categorize these criteria as supporting necessity order. We argue that evaluating attribution methods only with perturbation curves, *e.g. Area Under Curve (or AUC)*, only discovers the tip of the iceberg and potentially can be problematic. A toy model is shown in Example 1 to elaborate our concerns.

**Example 1.** *Consider a model $M(\mathbf{x}) = max(x_1, x_2)$ that takes a vector $\mathbf{x}$ with three features $x_1, x_2, x_3 \in \{0, 1\}$ but only $x_1$ and $x_2$ are relavant to the computation. Given the input to the model is $x_1 = x_2 = x_3 = 1$, assume $A_1, A_2, A_3$ are three different methods and output the attribution scores $s_1, s_2, s_3$ shown in Table 1 for each input feature $x_1, x_2, x_3$, respectively.*

| | $s_1$ | $s_2$ | $s_3$ |
|---|---|---|---|
| $A_1$ | $1/6$ | $1/3$ | $1/2$ |
| $A_2$ | $2/3$ | $0$ | $1/3$ |
| $A_3$ | $2/3$ | $1/3$ | $0$ |

Table 1. $s_1, s_2, s_3$ are attribution scores for $x_1, x_2, x_3$ computed by $A_1, A_2, A_3$, respectively.

*We apply zero perturbation to the input which means we set features to 0. The AOPC evaluation for these three attribution methods is shown in Fig 3. Using the conclusion from [? ] that higher AOPC scores suggest higher relativity of input features highlighted by an attribution method, Fig 3 shows pixels highlighted by $A_3$ are more relative ? ] to prediction than $A_1$ and $A_2$, as expected. However, $A_2$ and $A_1$ are considered as showing same level of relativity under the AOPC measurement even though $A_2$ succeeds in discovering $x_1$ is more relevant than $x_3$, whereas $A_1$ believes $x_3$ is more relevant than both $x_1, x_2$.*

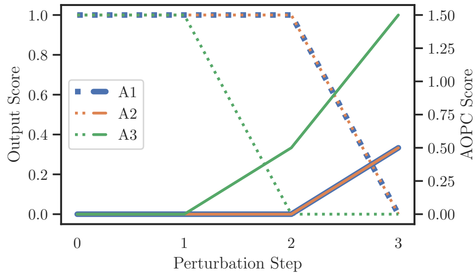

Figure 3. Comparing attribution methods $A_1$, $A_2$ and $A_3$ by applying zero perturbation. Dash lines are the change of model's output at each perturbation step (only one feature is set to 0 at each step). Solid lines are the changes of AOPC scores. $A_2$ and $A_3$ are overlapping with each other in this exmaple.

Another set of criteria instead stipulate that positively attributed features should stand on their own independently of non-important features. An example of this criterion is *Average % Drop* [? ] in support of GradCam++ that measures the change of class score by presenting only pixels highlighted by an attribution only (non-important pixels are ablated). Another example is the *LEast Relevant Features first (LeRF)* [? ] that removes the features with least high attribution scores. first. We will say these two criteria support sufficiency order (definition to follow).

Rethinking the concept of relativity, we believe both necessity and sufficiency can be treated as different types of relativity. In Example 1, neither $x_1$ nor $x_2$ is a necessary feature individually because the output will not change if any one of them is absent. However, both $x_1$ and $x_2$ are sufficient features, with either of which, the model could produce the same output as before. Besides, $A_2$ succeeds in placing the order of sufficient feature $x_1$ in front of the non-sufficient feature $x_3$ but $A_2$ fails, while AOPC(or AUC) is unable to discover the success.

Other evaluation criteria exist, like *sensitivity-n* [**?** ] and sanity check [**?** ], will be discussed in Section 5.

## 3. Methods

To tame the zoo of criteria, we organize and decompose them into two aspects: (1) *ordering* imposes conditions under which an input should be more important than another input in a particular prediction, and (2) *proportionality* further specifies how attribution should be distributed among the inputs. We elaborate on ordering criteria in Section 3.2 with instantiating in Section 3.3 and Section 3.4. We describe proportionality in Section 3.5. We begin with the logical notions of necessity and sufficiency as idealized versions of ablation-based measures described in Section 2. We introduce our notations in this paper before any further discussion,

### 3.1. Notation

Consider a model $y = f(\mathbf{x})$ and an attribution method $A$, it computes a set of attribution scores $s_1, s_2, ..., s_n$ for each pixel $x_1, x_2, ..., x_n$ in the input image $\mathbf{x}$ attributing a given class [1]. We permute the pixels into a new ordering $\pi_A(\mathbf{x}) = [x'_1, x'_2, ..., x'_n]$ so that $s'_1 \geq s'_2 \geq ... \geq s'_n$. We take the subset $\pi^+_A(\mathbf{x})$ of $\pi_A(\mathbf{x})$ so that $\pi^+_A(\mathbf{x})$ has the same ordering as $\pi_A(\mathbf{x})$ but only contains pixels with positive attribution scores. Let $R_i(\mathbf{x}, \pi)$ be the output of the model with input $\mathbf{x}$ where pixels $x'_1, x'_2, ..., x'_i \in \pi$ are perturbed from the input by setting $x'_1 = x'_2 = ... = x'_i = b$, where $b$ is a baseline value for the image (typically $b = 0$). Also, let $\mathbf{x}_b$ be the the baseline input image where all the pixels are filled with the baseline value $b$. Therefore, we have the the original output $y_0 = f(\mathbf{x})$ and the baseline output $y_b = f(\mathbf{x}_b)$.

### 3.2. Logical Order

The notions of necessity and sufficiency are commonly used characterizations of logical conditions. A necessary condition is one without which some statement does not hold. For example, in the statement $P_1 = A \wedge B$, both $A$ and $B$ are necessary conditions as each independently would invalidate the statement were they be made false. On the other hand, a sufficient condition is one which can independently make a statement true without other conditions being true. In more complex statements, no atomic condition may be necessary nor sufficient though compound conditions may. In the statement $P_3 = (A \wedge B) \vee (C \wedge D)$, none of $A, B, C, D$ are necessary nor sufficient but $(A \wedge B)$ and $(C \wedge D)$ are sufficient. As we are working in the context of input attributions, we relax and order the concept of necessity and sufficiency for atomic conditions (individual input pixels).

**Definition 1** (Logical Necessity Ordering)**.** *Given a statement $P$ over some set of atomic conditions, and two orderings $a$ and $b$, both ordered sets of the conditions, we say $a$ has better necessity ordering for $P$ than $b$ if:*

$$\min_i \left( \{a_k\}_{k \geq i} \not\models P \right) \leq \min_i \left( \{b_k\}_{k \geq i} \not\models P \right) \quad (1)$$

**Definition 2** (Logical Sufficiency Ordering)**.** *Likewise, $a$ has better sufficiency ordering for $P$ than $b$ if:*

$$\min_i \left( \{a_k\}_{k \leq i} \models P \right) \leq \min_i \left( \{b_k\}_{k \leq i} \models P \right) \quad (2)$$

A better necessity ordering is one that invalidates a statement $P$ by removing the shorter prefix of the ordered conditions while a better sufficiency ordering is the one that can validate a statement using the shorter prefix.

### 3.3. Necessity Ordering (N-Ord)

Unlike logical statements, numeric models do not have an exact notion of a condition (feature) being present or not. Instead, inputs at some baseline value or noise are viewed as having a feature removed from an input. Though this is an imperfect analogy, the approach is taken by every one of the measures described in Section 2 that make use of perturbation in their motivation. Additionally, with numeric outputs, the nuances in output obtain magnitude and we can longer describe an attribution by a single index like the minimal index of Definitions 1 and 2. Instead we consider an ideal ordering as one which drops the numeric output of the model the most with the least number of inputs ablated.

We refer the AUC measurement [**? ? ?** ] and MoRF [**?** ] as means to measure the Necessity Ordering (N-Ord). Denote $N_o(\mathbf{x}, A)$ as N-Ord score given a input image $\mathbf{x}$ and an attribution method $A$. Rewrite AUC using the notation in Section 3.1:

$$N_o(\mathbf{x}, A) = \frac{1}{M+1} \sum_{m=0}^{M} R^m_0(\mathbf{x}, A) \quad (3)$$

where $R^m_0(\mathbf{x}, A) = max\{R_m(\mathbf{x}, \pi^+_A(\mathbf{x})) - y_b, 0\}$ and $M$ is the total number of pixels in $\pi^+_A(X)$. We include $max$ to clip scores below the baseline output. According to Definition 1, we have the following proposition.

**Proposition 1.** *An attribution method $A_1$ shows a (strictly) better Ordering Necessity than another method $A_2$ given an input image $\mathbf{x}$ if $N_o(\mathbf{x}, A_1) < N_o(\mathbf{x}, A_2)$*

[1] we omit the notation of the class of interest for the simplicity in the rest of the paper

As discussed in Section 2.3, N-Ord only captures whether more necessary pixels, are receiving higher attribution scores. We argue that attribution methods should also be differentiated by the ability of highlighting sufficient features. To evaluate whether more sufficient pixels are receiving higher attribution scores, we propose Sufficiency Ordering as a complementary measurement.

### 3.4. Sufficiency Ordering (S-Ord)

We believe LeRF [? ] is a related means of measuring the Sufficiency. Sufficiency Ordering measures the score increase as we keep adding important features into a baseline input. Use the notation in Section 3.1 and let $R'_i(\mathbf{x}_b, \pi)$ be the model's output with $\mathbf{x}_b$ where $x'_1, x'_2, \cdots, x'_i \in \pi$ are added to the baseline image $\mathbf{x}_b$. Denote $S_o(\mathbf{x}, A)$ as S-Ord score given a input image $\mathbf{x}$ and an attribution method $A$.

$$S_o(\mathbf{x}, A) = \frac{1}{M+1} \sum_{m=0}^{M} R_0^{m'}(\mathbf{x}, A) \tag{4}$$

where $R_0^{m'}(\mathbf{x}, A) = min\{R'_m(\mathbf{x}_b, A), R'_M(\mathbf{x}_b, \pi_A^+(\mathbf{x}))\} - y_0$, $M$ is the number of pixels in $\pi_A^+(\mathbf{x})$. We include $min$ to clip scores above the original output. According to Definition 2, we have the following proposition.

**Proposition 2.** *An attribution method $A_1$ shows (strictly) better Ordering Sufficiency than another method $A_2$ given an input image $X$ if $S_o(\mathbf{x}, A_1) > S_o(\mathbf{x}, A_2)$.*

N-Ord and S-Ord together provides a more comprehensive evaluation for an attribution method. In Section 3.5, we are going to discuss the disadvantages of only using N-Ord or S-Ord and propose Proportionality as a refinement to the ordering analysis.

### 3.5. Proportionality

N-Ord and S-Ord do not incorporate the attribution scores beyond producing an ordering. This can be an issue toward an accurate description of feature necessity or sufficiency. For example, consider a toy model $M(x_1, x_2) = 2x_1 + x_2$ and let the inputs variables be $x_1 = x_2 = 1$. Any attribution methods that assign higher score for $x_1$ than $x_2$ produces the identical ordering $\pi(x_1, x_2) = [x_1, x_2]$, even one could overestimate the degree of necessity (or sufficiency) of $x_1$ by assigning it with much higher attribution scores. With *linear agreement* [? ], scores for $x_1$ and $x_2$ are more reasonable if their ratio is close to 2:1. Explaining a decision made by a more complex model only using ordering of attributions may overestimate or underestimate the necessity (or sufficiency) of an input feature. Therefore, We propose Proportionality as a refinement to quantify the necessity and sufficiency in complementary to the ordering measurement.

**Definition 3** (Proportionality-k for Necessity)**.** *Consider two positive number $n_1, n_2$ and an attribution method $A$. Use notations in Section 3.1 and let $\hat{\pi}_A^+(\mathbf{x})$ be a reversed ordering of $\pi_A^+(\mathbf{x})$. Proportionality-k for Necessity is measured by*

$$N_p^k(\mathbf{x}, A) = |R_{n_1}(\mathbf{x}, \pi_A^+(\mathbf{x})) - R_{n_2}(\mathbf{x}, \hat{\pi}_A^+(\mathbf{x}))| \tag{5}$$

*under the condition $\sum_i^{n_1} s_i = \sum_j^{n_2} s_j = kS(A, \mathbf{x}), s_i \in \pi_A^+(\mathbf{x}), s_j \in \hat{\pi}_A^+(\mathbf{x}), k \in [0, 1]$. $R_i(\mathbf{x}, \pi)$ uses the same definition in (3), and $S(\mathbf{x}, A)$ is the sum of total positive attribution scores.*

**Explanation of Definition 3** the motivation behind Proportionality-k for Necessity is that: given a group of pixels ordered with their attribution scores, there are different ways of distributing scores to each feature while the ordering remains unchanged. An optimal assignment is preferred that features receive attribution scores proportional to the output change if they are modified accordingly. In other words, given any two subsets of pixels $\pi_1$ and $\pi_2$. with total attribution scores sum to $S_1$ and $S_2$, are perturbed, the change of output scores $R(\mathbf{x}, \pi_1)$ and $R(\mathbf{x}, \pi_2)$ should satisfy $R(\mathbf{x}, \pi_1)/R(\mathbf{x}, \pi_2) = S_1/S_2$. This property is demanded because the same share of attribution scores should account for the same necessity or sufficiency. If we restrict the condition to $S_1 = S_2$, the difference between $R(\mathbf{x}, \pi_1)$ and $R(\mathbf{x}, \pi_2)$ becomes an indirect measurement of the proportionality. For the measurement of Necessity, we further restrict that $\pi_1$ is perturbed from the pixel with the highest attribution score first and $\pi_2$ is perturbed from the one with lowest attribution score first, in accordance with the setup in N-Ord. Therefore, a smaller difference $N_p^k(\mathbf{x}, A)$ shows better Proportionality-k for Necessity

**Proposition 3.** *An attribution method $A_1$ shows better Proportionality-k for Necessity than method $A_2$ if $N_p^k(\mathbf{x}, A_1) < N_p^k(\mathbf{x}, A_2)$*

A similar requirement for attribution method is *completeness* discussed by [? ] and its generalization *sensitivity-n* discussed by [? ]. *completeness* requires the sum of total attribution scores to be equal to the change of output compared to a baseline input, and *sensitivity-n* requires any subset of $n$ pixels whose summation of attribution scores should be equal to the change of output compared to the baseline if pixels in that subset are removed. When $n$ is the total number of pixels in the input image, *sensitivity-n* reduces to *completeness*. The relationships between *sensitivity-n* and Proportionality-k for Necessity are discussed as follows:

**Proposition 4.** *If an attribution method $A$ satisfies both sensitivity-$n_1$ and sensitivity-$n_2$, then $N_p^k(\mathbf{x}, A) = 0$ under the condition if $\sum_i^{n_1} s_i = \sum_j^{n_2} s_j = kS(\mathbf{x}, A), s_i \in \pi_A^+(\mathbf{x}), s_j \in \hat{\pi}_A^+(\mathbf{x}), k \in [0, 1]$ , but not vice versa.*

The proof for Proposition 4 and can be found in Appendix 1. We further contrast our method with *sensitivity-n* in Section 5. Integrating *proportionality* with all possible shares of attribution scores, we define the Total Proportionality for Necessity (TPN):

**Definition 4** (Total Proportionality for Necessity)**.** *Given an attribution method A and an input image* $\mathbf{x}$*, The Total Proportionality for Necessity is measured by*

$$N_p(\mathbf{x}, A) = \frac{1}{r y_0} \int_0^1 N_p^k(\mathbf{x}, A) dk \tag{6}$$

*where* $r = \min\{y_b / R_M(\mathbf{x}, \pi_A^+(\mathbf{x})), 1\}$[2]*.* $y_0$ *is used as a normalizer and* $M$ *is the total number of elements in* $\pi_A^+(\mathbf{x})$*, therefore,* $r = 1$ *if removing all elements in* $\pi_A^+(\mathbf{x})$ *drops the score to the baseline. Revisit the Section 3.1 for notations if needed.*

**Explanation for Definition 4** $N_p(\mathbf{x}, A)$ is the area between two perturbation curves one starting from the pixels with highest attribution scores and the other with a reversed ordering. The difference from Necessity Ordering is that $N_p(\mathbf{x}, A)$ is measured against the share of attribution scores (the value of $k$) instead of the share of pixels in the $N_o(\mathbf{x}, A)$. On the other side, perturbations on non-necessary features may not change the output at all and we penalize an attribution method that guides us to do so with the ratio $r$ compared to the baseline. Generalizing Proposition 3, we argue:

**Proposition 5.** *An attribution method* $A_1$ *shows better Total Proportionality for Necessity than method* $A_2$ *if* $N_p(\mathbf{x}, A_1) < N_p(\mathbf{x}, A_2)$

Under the similar construction, we have the following definition of Proportionality-k for Sufficiency and Total Proportionality for Sufficiency (TPS):

**Definition 5** (Proportionality-k for Sufficiency)**.** *Consider two positive number* $n_1, n_2$ *and an attribution method A. Use notations in Section 3.1 and let* $\hat{\pi}_A^+(\mathbf{x})$ *be a reversed ordering of* $\pi_A^+(\mathbf{x})$*. Proportionality-k for Sufficiency is measured by*

$$S_p^k(\mathbf{x}, A) = |R'_{n_1}(\mathbf{x}_b, \pi_A^+ \mathbf{x})) - R'_{n_2}(\mathbf{x}_b, \hat{\pi}_A^+(\mathbf{x}))| \tag{7}$$

*under the condition* $\sum_i^{n_1} s_i = \sum_j^{n_2} s_j = kS(\mathbf{x}, A), s_i \in \pi_A^+(\mathbf{x}), s_j \in \hat{\pi}_A^+(\mathbf{x}), k \in [0, 1]$*.* $R'_i(\mathbf{x}, \pi)$ *reuses the definition in* (4)*;* $S(\mathbf{x}, A)$ *is the sum of total positive attribution scores.*

We want the difference $S_p^k(\mathbf{x}, A)$ as small as possible since the same share of attribution scores should reflect same sufficiency. Therefore, we have the following proposition:

**Proposition 6.** *An attribution method* $A_1$ *shows better Proportionality-k for Sufficiency than method* $A_2$ *if* $S_p^k(\mathbf{x}, A_1) < S_p^k(\mathbf{x}, A_2)$

**Definition 6** (Total Proportionality for Sufficiency)**.** *Given an attribution method A and an input image* $\mathbf{x}$*, The Total Proportionality for Sufficiency is measured by*

$$S_p(\mathbf{x}, A) = \frac{1}{r' y_0} \int_0^1 S_p^k(\mathbf{x}, A) dk \tag{8}$$

*where where* $r' = \min\{R'_M(\mathbf{x}, \pi_A^+(\mathbf{x}))/y_0, 1\}$*.* $y_0$ *is used as a normalizer and* $M$ *is the total number of elements in* $\pi_A^+(\mathbf{x})$*, therefore,* $r' = 1$ *if adding all elements in* $\pi_A^+(\mathbf{x})$ *increases the score to the original output. Refer to Section 3.1 and 3.4 for details about the notation.*

Similarly, $S_p(\mathbf{x}, A)$ is the area between curves of model's output change by adding pixels to a baseline input with the highest attribution scores first or by the lowest first. The ratio $r'$ penalizes the false postive situation when adding all pixels with positive scores does not increase the output significantly. Finally, we have

**Proposition 7.** *An attribution method* $A_1$ *shows better Total Proportionality for Sufficiency than another method* $A_2$ *if* $S_p(\mathbf{x}, A_1) < S_p(\mathbf{x}, A_2)$

In summary, we differentiate and describe the Necessity Ordering (N-Ord) and Sufficiency Ordering (S-Ord) from previous work and propose Total Proportionality for Necessity (TPN) and Total Proportionality for Sufficiency (TPS) as refined evaluation criteria for necessity and sufficiency. We then apply our measurement to explain the prediction results from an image classification task in the rest of the paper.

## 4. Evaluation

### 4.1. Implementation of Proportionality

To compute TPN for each single input, we ablate a subset of input features. Unlike in Ordering, we do not ablate a certain number of features, we ablate a subset of features with a certain share of attribution. The share of attribution scores $k$ goes from 0 to 1. We generate the ablation curves by removing features with highest scores first and those with the lowest scores first and measure the area between these the curves. Optimal TPN will have area 0 as discussed in the previous sections. The analogous is done for TPS.

### 4.2. Evaluate on the datasets

We evaluate N-Ord, S-Ord, TPN, and TPS on a 9600 image sample of ImageNet [**?** ] on VGG16 models. We evaluate all methods motioned in Section 3. Details of the model and attribution methods can be found in the Appendix 3.

[2] We clip the scores below 0 and add a small positive number $\epsilon$ to the denominator to ensure the numerical stability.

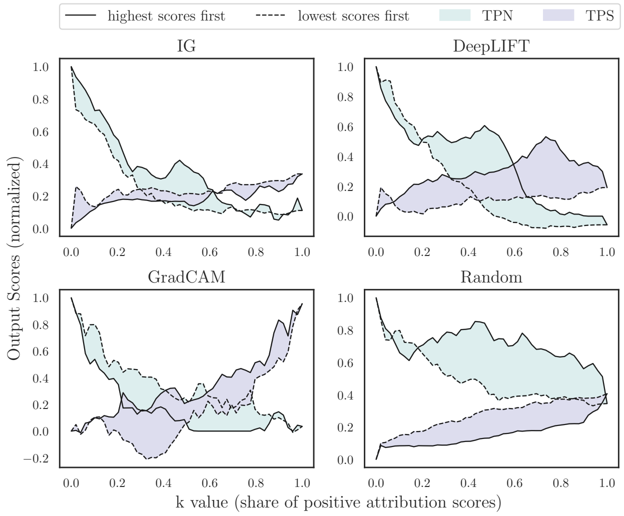

Figure 4. TPN and TPS curves for the input of Fig 2. The area represents the scores before the penalty factors $r$ and $r'$. As a baseline, the Random method randomly assigns attribution scores. This figure is better viewed in color.

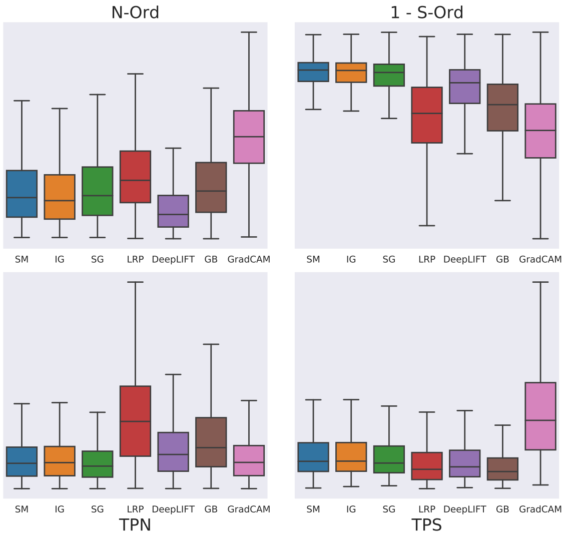

Figure 5. The boxplot for different attribution methods evaluated with four crietria aforementioned on 9600 images from ImageNet with VGG16 model. Since only the higher S-Ord score indicates better Sufficiency Ordering, we use 1− S-Ord to accord with other criteria **so the lower scores will indicate better performance on all criteria shown above**.

The boxplot in Fig 5 summarizes the results. Note that we plot 1− S-Ord instead the original S-Ord score so that that conclusion that lower scores represent better performance holds for all subfigures. On the ImageNet and VGG model, DeepLIFT shows relatively better Necessity Ordering while GradCAM shows relatively better Sufficiency Ordering. Saliency Map, Integrated Gradient, and Smooth Gradient are all showing slightly better proportionality for necessity while LRP and GB are showing slightly better proportionality for sufficiency. No evaluated method is significantly better than others on all criteria simultaneously.

**Saliency Map** performs not bad in necessity for both ordering and *proportionality* compared to sufficiency; the features with highest scores assigned by Saliency Map may not be sufficient for the decision making process, *e.g*. the pool is highlighted by Sailency Map in Figure 2 but the model may not make a mistake when only the pool is present. One possible reason is vanishing gradient causes a loss of signal. **Integrated Gradient and DeepLIFT** are two both motivated by the vanishing gradient problem of Saliency Map and both achieve better Necessity Ordering. However, they do not show significantly better proportionality in both necessity and sufficiency. The reason behind this we assume is that the *Summation-to-Delta* requirement only guarantees that sum of attribution scores for all features equals to the change of output, while any other share of attribution scores does not cause equivalent change to the output, so the proportionality is not improved. Similar conclusions are also discussed by *sensitivity-n* [**?** ] **Smooth Gradient** shows lower inter-quartile range in both TPN and TPS compared to Saliency Map. Computing the expectation of the Salience Map in a distribution of inputs does not resolve the possible vanishing gradients issue for each input in the distribution; however, at the same activation unit, *e.g*. ReLU, an input's gradient signal is blocked by the flat negative region but its neighbor's gradient signal can get unblocked. It may help to explain the improvement Smooth Gradient shows in the experiments. **GradCAM** is the best choices for Sufficiency Ordering regardless of it doing poorly in proportionality for the sufficiency – the attribution scores may not reflect actual sufficiency. The result is not surprising as the upsampling process in GradCAM does not relate to any axiom that guarantees to produce pixel-level proportional scores.

On the contradictory, we can not make instructive comment on the following two attribution methods: **Guided Backpropagation** shows better sufficiency on ordering and proportionality compared to the necessity. We consider it as a good method to reveal the sufficient features, however, as **?** ] points out GB lacks fearfulness to the model by behaving poorly in the *sanity check*. Therefore, we leave the understanding of GB as a future work. On the other hand, **Layer-wise Relevance Propagation** is the one we will not make much strong conclusion as well since there are many rules in LRP and only one of them, $\alpha 2\beta 1$-LRP (see Appendix II), is tested. But specifically, for $\alpha 2\beta 1$-LRP, it shows good sufficiency on both criteria, which increase our confidence to interpret the result of $\alpha 2\beta 1$-LRP as identifying sufficient features in the input space.

### 4.3. Evaluate with one instance

All metrics can be applied to a single input and the interpretation using all winners for each criteria can provide more insights about the model. For example, in Fig 6, we interpret that the body of a dog is necessary to the `English springer` class and only providing the body the model may not consider it is a dog, the sufficient feature is its head. Consulting different attributions and interpret with the winners can give more comprehensive understandings. More exmaples are included in Appendix III.

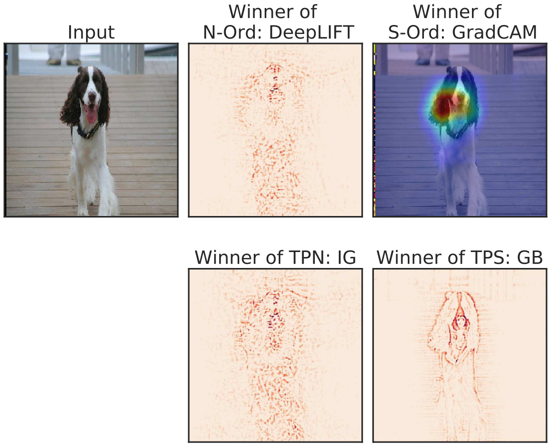

Figure 6. An example of interpreting the model's predictions with winners on different criteria.

## 5. Related Work

To the best of our knowledge, we are the first one describing the concepts of necessity and sufficiency in attributions where similar work may only touch the surface of either necessity or sufficiency but not both. Our work is partially motivated by *smallest sufficient region* (SSR) and *smallest destroying region* (SDR) [**?** ] where the authors aim to propose a region that either increase or decrease the model's output most. Though SSR and SDR only capture the spatial location in the image, they do not incorporate the feature contributions as scores.

We view our work as a variant of **sensitivity** evaluation of an attribution's magnitude (instead of just order). A related concept is *quantitative input influence* by **?** ] (even though its authors do not target on deep neural networks). *sensitivity (a)(b)* [**?** ] provides the basis of discussion and *sensitivity-n* [**?** ] imposes stricter requirements. The mathematical connection between *proportionality* and *sensitivity-n* is discussed in Section 3.5. *proportionality* approaches the sensitivity from a view that, regardless of the number of pixels, same share of attribution should account for same change to the output, while *sensitivity-n* requires removing $n$ pixels should change the output by the amount of total attribution scores of that $n$ pixels. *sensitivity-n* is a non quantitative criterion while *proportionality* is numerical and can be used to compare different methods under two orderings: the necessity and sufficiency.

## 6. Conclusion

In this work, we summarized existing evaluation metrics for attribution methods and categorized them into two logical concepts, necessity and sufficiency. We then demonstrated realizable criteria to quantify necessity and sufficiency with an analysis focused on ordering and its refinement, proportionality. We evaluated attribution methods against our criteria and listed the best methods for each criteria. We discovered that some attribution methods excel in necessity or sufficiency, but none is winner for both.

The logical concepts of necessity and sufficiency are generally mutually exclusive and our analogues show the same based on our results: no method is universally optimal for both necessity and sufficiency. While this means we cannot endorse one method over others, the techniques we present provide additional interpretability tools to data scientist who can use our measures to select the attribution appropriate to the task at hand. When debugging a model for identifying traffic stop signs, an analyst can select for methods with greater necessity to determine whether the model has learned spurious correlates, e.g., the pole holding up the sign. A "necessary" pole would lead to false negatives (stop signs not on poles) while a "sufficient" one would only indicate potential false positives (poles without stop signs) which, though also problematic, are not as dangerous as false negatives in this case. The increased basis with which to interpret attribution will hopefully lead to a fuller understanding of model behaviour.

**Acknowledgements** This work was developed with the support of NSF grant CNS-1704845 as well as by DARPA and the Air Force Research Laboratory under agreement number FA8750-15-2-0277. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes not withstanding any copyright notation thereon. The views, opinions, and/or findings expressed are those of the author(s) and should not be interpreted as representing the official views or policies of DARPA, the Air Force Research Laboratory, the National Science Foundation, or the U.S. Government.

We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan V GPU used for this research.