EVALUATION OF ATTRIBUTION EXPLANATIONS WITHOUT GROUND TRUTH

Anonymous authors
Paper under double-blind review

ABSTRACT

This paper proposes a metric to evaluate the objectiveness of explanation methods of neural networks, *i.e.,* the accuracy of the estimated importance/attribution/saliency values of input variables. This is crucial for the development of explainable AI, but it also presents significant challenges. The core challenge is that people usually cannot obtain the ground-truth value of the attribution of each input variable. Thus, we design a metric to evaluate the objectiveness of the attribution map without ground truth. Our metric is used to evaluate eight benchmark methods of attribution explanations, which provides new insights into attribution methods. *We will release the code when the paper is accepted.*

1 INTRODUCTION

Nowadays, many methods have been proposed to explain deep neural networks (DNNs) in a post-hoc manner. In this research, we limit our attention to existing attribution methods of estimating the *importance/attribution/saliency* of input variables or neural activations in an intermediate layer to the network output (Lundberg & Lee, 2017; Ribeiro et al., 2016; Bach et al., 2015).

In this paper, we aim to evaluate the objectiveness of eight existing attribution methods, rather than propose a new attribution method to explain the DNN. The motivation of evaluating the objectiveness of attribution methods is that we cannot ensure the inference logic of the DNN always fit the logic of human cognition. Specifically, if the attribution value of an input variable (*e.g.*, a pixel in an image, a word in a sentence) is estimated twice of another input variable's attribution, the goal of this research is to develop a metric to check whether the first input variable really makes exactly twice influence on the prediction *w.r.t.* the second input variable. This is fully different from empirically checking whether the attribution map fits the human cognition.

However, people usually cannot obtain the ground-truth logic used by the DNN for inference. First, there is a large gap between visually reasonable attributions and the real inference logic in a DNN (*e.g.,* Ren et al. (2021) found that human intuition was different from causal patterns mined from the DNN). As Figure 1 shows, different methods generate dramatically different attribution maps, although most attribution maps look reasonable. To this end, Cui et al. (2019); Yang et al. (2019) evaluated whether the attribution map looks reasonable to human users in a qualitative manner. However, the intuitive examination is not powerful enough to quantify the potential bias of attributions.

Second, although the typical way of evaluation based on masking input variables in a certain order (Samek et al., 2017; Ancona et al., 2018) has been widely accepted as a standard, we prove that this evaluation method has a mathematical flaw. Specifically, Deng et al. (2021) proved that most attribution methods can be explained as the re-allocation effects of the interaction between input variables to these variables. We further prove that the sequential masking strategy causes an unbalanced allocation of interaction effects, and hurts the fairness of the evaluation.

Third, some well-known studies (Yang & Kim, 2019; Camburu et al., 2019) constructed specific datasets with empirically determined "ground-truth attributions," but we conduct experiments to demonstrate that such datasets cannot guarantee to fully represent all true inference logic of a DNN for evaluation. It is because the DNN usually mistakenly learns some chaotic features, which are not supposed to be learned.

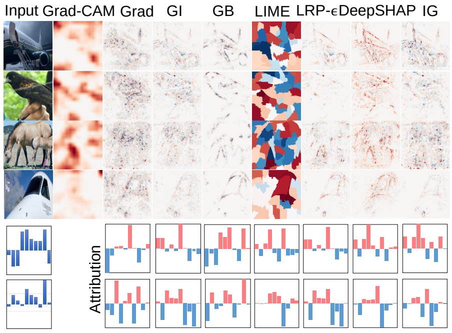

Figure 1: Conflicting attribution maps generated by different methods. Different attribution methods generate different attribution maps for the same image. The last two rows show results on the TV news dataset (Vyas et al., 2014). Furthermore, in experiments, we show that explanations should not be evaluated using ground truth labeled according to human cognition. Appendix A provides more attribution maps.

Table 1: Review of attribution methods

Method	What to explain	Limitations
Grad-CAM	Attri. distribution at an intermediate layer	Usually explaining features at high layers
Grad	Attri. of input variables	–
GI	Attri. of input variables	–
IG	Attri. of input variables	–
GB	Attri. of input variables	Requirement to use ReLU as non-linear layers
Shapley Value	Attri. of input variables	NP-hard problem
DeepSHAP	Attri. of input variables	Only be applied to certain architectures, which are designed with specific backward rules.
LIME	Super-pixel level attribution	Analysis at the super-pixel level, rather than the pixel level
LRP-ϵ	Attri. of input variables	Only be applied to certain architectures, whose relevance propagation rules of all layers should be defined.

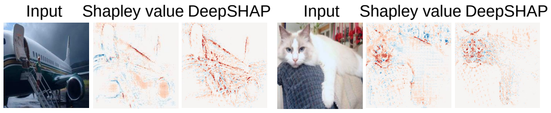

Figure 2: Difference between attribution maps generated by DeepSHAP and attribution maps of relatively accurate Shapley values computed by conducting over 20,000 times of sampling.

How to evaluate the objectiveness of the explanation? Defining theoretical rules to examine the objectiveness of explanations is still an open problem. To this end, Shapley (1953); Deng et al. (2021) defined game-theoretic axioms for attributions to identify better attribution methods. For example, Shapley (1953) proposed *linearity, dummy, symmetry, and efficiency axioms* for objective attributions. Besides, Deng et al. (2021) proved the *interaction distribution axiom*. To this end, *we admit that the above five axioms are not the only golden standard for objective attributions*. We welcome new axioms, but we still believe that these five axioms are good enough to evaluate the objectiveness of attribution methods. Please see Section 2 for more discussions.

However, directly computing attributions that satisfy the above five axioms for evaluation is **not practical** (actually, the Shapley value satisfies such axioms). First, the computational cost of accurately computing the Shapley value is NP-hard. Second, the approximation of the Shapley value (Lundberg & Lee, 2017) has very large errors. The gap between the accurate Shapley value and the approximated Shapley value is usually even larger than the gap between other explanations and accurate Shapley values, which makes this evaluation metric unreliable.

Therefore, we neither compute a more accurate attribution, nor directly evaluate the attribution of each input variable. Instead, we design a new metric to evaluate the bias of the distribution of attributions. This metric can be accurately computed without much computational cost. More crucially, this metric can be derived from the aforementioned axioms (see Appendix C). *Besides, experiments show that this evaluation does not have a consistent partiality to Shapley-value-based explanations (e.g. DeepSHAP (Lundberg & Lee, 2017)) in all DNNs*, owing to the gap between the Shapley-value-based explanation and the accurate Shapley value (see Figure 2).

Evaluation on pixels or super-pixels? How to determine the basic unit (*e.g.* the elementary concept) of the explanation is another open problem without a trustworthy solution. Therefore, to simplify the story, this paper only focuses on the objectiveness of the pixel-wise attribution.

Contributions of our paper can be concluded as follows. (1) We propose a standard metric based on game theory to evaluate the bias of the estimated attributions without knowing ground-truth explanations. (2) Instead of examining the attribution of each input variable, the new evaluation metric towards the bias of the attribution ensures both computational efficiency and the accuracy.

2 Evaluation metrics of explanations

Axioms for objectiveness. The objectiveness of explaining a DNN is a key issue for attribution methods, but it is still an open problem. In recent years, many studies have attempted to explore the trustworthiness of an explanation result. A classical way is to define various axioms to examine the trustworthiness of attribution maps. Shapley (1953) proposed *linearity, dummy, symmetry, and efficiency axioms*, and considered that trustworthy attributions should satisfy such axioms (see Appendix E for details). Furthermore, Deng et al. (2021) further proposed the *interaction distribution axiom* as shown in Theorem 1, and used interactions between input variables as a unified explanation for the rationale of different attribution methods.

The composition of the five axioms has been widely considered as a typical requirement for the objective explanation of DNNs (Shapley, 1953; Ancona et al., 2019; Lundberg & Lee, 2017), although we admit that the five axioms are not the only golden standard to evaluate the objectiveness of explanations, and we welcome further axioms.

Details about axioms and the Shapley value. Shapley (1953) and Harsanyi (1963) proved that the Shapley value was the unique solution that satisfied *linearity, dummy, symmetry, efficiency axioms*, and *the interaction distribution axiom* (see Theorem 1). The inference of a DNN can be regarded as a game with n players $\Omega = \{1, 2, \cdots, n\}$. Each player i is an input variable (*e.g.* a pixel/local patch/super-pixel). The DNN does not use each input variable (player) individually. Instead, a set of input variables (players) $T \subseteq \Omega$ may cooperate with each other and form a certain inference pattern (a coalition) to affect the network output (*i.e.,* pursuing a high reward). In this case, the DNN is taken as the game F, $F : 2^{\Omega} \to \mathbb{R}$. $F(T)$ represents the output of the DNN when only input variables in T are present, and all other variables in $\Omega \setminus T$ are masked. Specifically, $F(T)$ is computed on the sample, in which variables in $\Omega \setminus T$ are replaced with their baseline values. The baseline value is computed as the average input value over all input samples (Ancona et al., 2019). $F(\Omega) - F(\emptyset)$ denotes the total reward of all players. An attribution method tries to allocate the network output (the total reward) to each input variable $i \in \Omega$ (each player) as its attribution $A_i \in \mathbb{R}$. The Shapley value of the player i can be computed as a specific attribution $A_i = A_i^{\text{shap},*}$, as follows.

$$A_i^{\text{shap},*} = \sum\nolimits_{T \subseteq \Omega \setminus \{i\}} \frac{|T|!(|\Omega| - |T| - 1)!}{|T|!} \left[F(T \cup \{i\}) - F(T)\right] = \mathbb{E}_r \left[\mathbb{E}_{|T|=r, T \not\ni i} \left[F(T \cup \{i\}) - F(T)\right]\right], \quad (1)$$

where $|\Omega|$ denotes the total number of input variables (players) in an input sample, and r denotes the number of input variables (players) in a sampled subset T.

Linearity axiom: If the reward of a game F satisfies $F(T) = G(T) + H(T)$, where G and H are another two games, then $A_i^{(F)} = A_i^{(G)} + A_i^{(H)}$.
Dummy axiom: If an input variable i satisfies $\forall T \subseteq \Omega \setminus \{i\}, F(T \cup \{i\}) = F(T) + F(\{i\})$, then $F(\{i\}) - F(\emptyset) = A_i$.
Symmetry axiom: If two input variables i and j satisfy $\forall T \subseteq \Omega \setminus \{i, j\}, F(T \cup \{i\}) = F(T \cup \{j\})$, then $A_i = A_j$.
Efficiency axiom: $\sum_{i \in \Omega} A_i = F(\Omega) - F(\emptyset)$.

Theorem 1 *The Shapley value satisfies the interaction distribution axiom (Harsanyi, 1963), i.e.,* $A_i^{\textit{shap},*} = \sum_{T \subseteq \Omega : i \in T} D(T)/|T|$*, where* $D(T)$ *represents the Harsanyi dividend (Harsanyi, 1963) of the set of players* T*. The Harsanyi dividend* $D(T)$ *measures the numerical utility created by the interaction patterns among exactly all players in* T*. In this way, the Shapley value uniformly allocates each interaction pattern* $D(T)$ *to all players in this pattern.*

Evaluating the bias of the attribution distribution, rather than each specific attribution value. Although we can consider the Shapley value as the ground-truth explanation *w.r.t.* the above five axioms, we do **NOT** directly compute accurate attributions that satisfy the axioms for evaluation, since it is impractical in real applications. First, the accurate computing based on Equation (1) is NP-hard. Second, although Lundberg & Lee (2017); Chen et al. (2018); Castro et al. (2009); Ancona et al. (2019) proposed various methods to approximate the Shapley value with a relatively low computational cost, both (Aas et al., 2019) and Figure 2 show that accurate Shapley values are dramatically different from the approximated Shapley value. In fact, the gap between the approximated Shapley value and the accurate Shapley value is sometimes even larger than the gap between other explanations and the accurate Shapley value.

Therefore, we propose a new metric to evaluate the bias of the attribution distribution, instead of directly evaluating the attribution accuracy of each input variable. The attribution bias is used to check whether a group of top-ranked input variables with the largest (or smallest) attributions are assigned with more/less importance than the truth. More crucially, because this metric does not require us to compute the accurate attribution value for each input variable, the computational cost is significantly reduced. Besides, because the attribution bias is derived on the basis of the Shapley value, it reflects the aforementioned five axioms. Experimental results show that our evaluation metric does not have an incorrect partiality to inaccurate approximation of Shapley values (the inaccuracy is illustrated in both (Aas et al., 2019) and Figure 2). This verifies the fairness of this metric.

To compute the attribution bias, we are given an input sample $I \in \mathbf{I}$. Let us consider the DNN F with a single scalar output $y = F(I)$. For DNNs with multiple outputs, existing methods (Shrikumar et al., 2016; Lundberg & Lee, 2017; Ribeiro et al., 2016; Bach et al., 2015; Binder et al., 2016) usually explain each individual output dimension independently. Let $\{a_i\}$ denote the attribution map estimated by a specific attribution method. We aim to evaluate the bias of $\{a_i\}$. People propose the *efficiency axiom*, *i.e.,* requiring the network output to equal the sum of attribution values (Ancona et al., 2019; Lundberg & Lee, 2017), in order to let the attributions act like causal factors for the model output, to some extent. However, not all explanation methods satisfy this property. To ensure fair comparisons, we use λ to normalize attributions, so as to make attributions fit the efficiency axiom without loss of generality. Details about λ will be discussed under Equation (3).

$$y = b + \sum\nolimits_{i\in\Omega} A_i, \qquad \text{s.t.} \quad A_i = \lambda a_i \tag{2}$$

where b denotes the network output given an *empty* input, in which all variables are masked, according to (Shapley, 1953); i denotes the index of each input variable in the input sample; Ω denotes the set of all input variables in the input sample. Since many attribution methods (Selvaraju et al., 2017; Simonyan et al., 2013) mainly compute relative values of attributions $\{a_i\}$, instead of a strict attribution map $\{A_i\}$, we use λ to bridge $\{A_i\}$ and $\{a_i\}$, which is independent with the index i.

Assumption 1 *The estimated attribution of each input variable can be assumed to follow a Gaussian distribution $A_i \sim \mathcal{N}(\mu_i, \sigma_i^2)$. Attribution distributions of different input variables can be further assumed to share a unified variance, i.e., $\sigma_1^2 \approx \sigma_2^2 \approx ... \approx \sigma_n^2$. This assumption faithfully reflects the truth in practice, which has been verified in experiments in Appendix M.*

Lemma 1 *Let A_i^{shap} denote the unbiased approximation of the Shapley value based on the sampling method (Castro et al., 2009), which is given by the right side of Equation (1). Just like in (Castro et al., 2009), A_i^{shap} is assumed to follows a gaussian distribution $\mathcal{N}(A_i^*, (\sigma^{shap})^2)$ (in fact, such an assumption fit the truth in practice, and is verified in Appendix M). A_i^* denotes the true Shapley value. Then, given a specific set of input variables S, we have $\frac{\sum_{i\in S} A_i^{shap}}{|S|} \sim \mathcal{N}(\frac{\sum_{i\in S} A_i^*}{|S|}, \frac{(\sigma^{shap})^2}{|S|})$.*

Evaluating the attribution distribution $\{a_i\}$ has two steps. *First, we sample input variables whose attributions are more likely to have large deviations.* To this end, we sample the set of input variables S with top-ranked high (or low) attributions. These input variables in S are more likely to be biased towards high (or low) attribution values. Meanwhile, from another perspective, considering the Gaussian distribution of $\{A_i\}$, the distribution of the sampled attribution values is close to the Gumbel distribution (Gumbel, 1935).

Second, we measure the bias of the sampled top-ranked attributions to evaluate the attribution distribution. Specifically, we propose Definition 1 to define the anchor value β_S as the average value of the sampled top-ranked attributions in S, in order to evaluate the bias of the distribution of attributions. According to Lemma 1, the anchor value β_S has much lower variance than the attribution A_i^{shap} of each variable.

Definition 1 *Let S contain input variables with the highest (or the lowest) attributions in the sample. $\beta_S = (\sum_{i\in S} A_i^{shap})/(|S|\|A^{shap}\|)$ denotes the anchor value of our metric on the set S. In this case, the difference between the average attribution value of input variables in S and the anchor value, $\left|(\sum_{i\in S} A_i)/(|S|\|A\|) - \beta_S\right|$, is defined as the bias of the attribution map, as follows.*

$$M_{bias} = \mathbb{E}_I\left[\left|\frac{\sum_{i\in S} A_i}{|S|\|A\|} - \beta_S\right|\right] = \mathbb{E}_I\left[\frac{1}{|S|}\left|\frac{\sum_{i\in S} a_i}{\|a\|} - \frac{\sum_{i\in S} A_i^{shap}}{\|A^{shap}\|}\right|\right] \tag{3}$$

where A_i^{shap} denotes the Shapley value approximated by sampling. $\|A^{shap}\|$ and $\|a\|$ are used for normalization. A small value of M_{bias} indicates the low bias of the attribution map. In the computation of M_{bias}, the parameter λ has been eliminated.

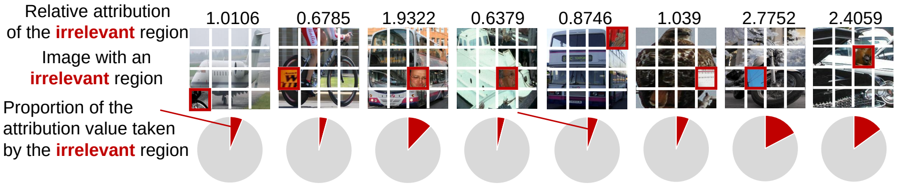

Figure 3: Disproof of the assumption that regions irrelevant to the classification had no contribution to the classification. Each image was divided into 4×4 regions. The red box indicates the region irrelevant to the target class, which was pasted into the image. Just like in (Yang & Kim, 2019), the AlexNet was trained on the dataset with irrelevant regions. Given the AlexNet, we computed accurate Shapley values of irrelevant regions via brute-force enumeration without approximation, and we found that the Shapley value of the irrelevant region was not zero, which demonstrated that it was untrustworthy to create the ground-truth explanation using irrelevant objects.

Can the above metric be used to estimate attributions or have partiality to Shapley-value-based methods? The proposed metric measures the bias of the distribution of attributions to evaluate attribution methods, rather than approximate the specific attribution of each input variable. Please see Appendix D for discussions. Besides, although the metric has the same theoretical foundation as the Shapley value, our metric has no partiality to the DeepSHAP, which fits the finding of the inaccuracy of DeepSHAP. For example, experimental results showed that LRP-ϵ outperformed DeepSHAP.

In addition, the proposed metric can also be used to evaluate the attribution of neural activations in the intermediate layer, such as those generated by Grad-CAM. In this case, we regard the target intermediate-layer feature as the input to compute A_i^{shap}, so as to implement the evaluation.

Unlike directly evaluating the accuracy of the attribution of each input variable, the metric for the attribution bias can be more accurately measured with much less sampling of testing samples (see Exp.1 in Section 4). Computing the accurate Shapley value is an NP-hard problem with the computational cost $\mathcal{O}(N \cdot 2^N)$, according to Equation (1). In comparison, the computational cost of the evaluation based on our metric is $\mathcal{O}(mN)$, where N denotes the number of variables in the input sample, and m is the sampling number of subsets T in Equation (1). Besides, Figure 5 shows that the estimated metric has been accurate enough for evaluation with 1000 sampling times.

For comparison, let us construct another evaluation metric as a baseline, which directly approximates relatively accurate attributions (i.e., approximated Shapley values A_i^{shap}) for evaluation. To this end, many approximation studies (Lundberg & Lee, 2017; Covert & Lee, 2021; Wang et al., 2022) have clear algorithmic bias, and cannot be used for evaluation. Thus, we still choose the unbiased sampling-based approximation (Castro et al., 2009). Proposition 1 shows that the anchor value of our metric β_S can be computed $|S|$ times more efficiently than the baseline metric A_i^{shap}. The efficiency of the anchor value β_S is verified in Figure 5 and is theoretically proved by Proposition 1.

Proposition 1 *(Proved in Appendix G) Let β_S be computed based on A^{shap}, which is approximated by sampling a total of m different contexts $T \subseteq \Omega$ according to Equation (1). Then, people need to sample $(|S| \cdot m)$ different contexts to compute $\hat{A}_i^{shap}$ to ensure $Var[\hat{A}_i^{shap}] = Var[\beta_S]$. $Var[\cdot]$ denotes the variance caused by the uncertainty of sampling.*

3 Related work, problems of previous evaluation metrics

Can we create ground-truth explanations according to human cognition for evaluation? Yang & Kim (2019); Camburu et al. (2019) have made a breakthrough in evaluation of attribution maps, *i.e.,* obtaining intuitive ground-truth explanations by creating synthetic datasets. These intuitive ground-truth explanations were used to evaluate the attribution map. Specifically, they pasted an irrelevant object *w.r.t.* the task into the image. Pixels of the irrelevant object were assumed to be assigned with zero attribution to the prediction.

However, we found that this assumption was not always convincing. A common sense is that the DNN does not make inferences in the same way as people (Jacob et al., 2021), so we conducted experiments to examine the above assumption. We followed the same settings to build up a synthetic

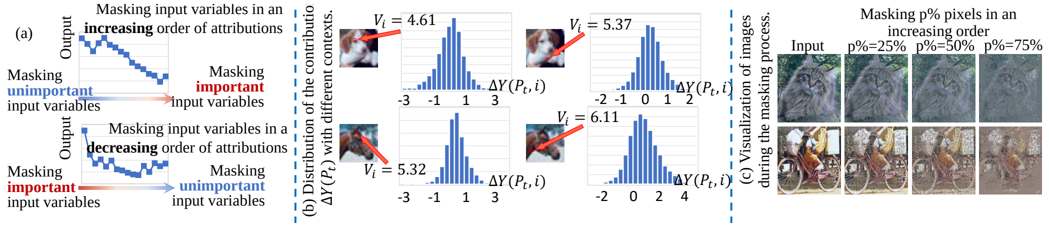

Figure 4: (a) Schematics of the evaluation based on ranked attributions. (b) Evaluating the trustworthiness of the evaluation method based on the ranked attributions. We randomly select the pixel i in the input image, and compute $\Delta Y(P_t, i)$ and $V(P_t)$ over different contexts P_t. We find that the contribution of the pixel to the output is unstable *w.r.t.* different contexts, which verifies the problem with evaluation based on the ranked attributions in Remark 1. (c) The masked pixels distribute over the whole image when we evaluate a specific attribution map. Therefore, the evaluation based on the removal of ranked pixels is conducted under specific a few contexts. In comparison, our evaluation metric is computed considering all contexts, which is more reliable according to Theorem 1.

dataset by modifying images in the Pascal VOC 2012 dataset (Everingham et al., 2010) to contain both relevant regions and irrelevant regions. We divided each image into 4×4 regions. Then, we randomly replaced a region with an image patch of 56×56 pixels cut from an image in a different class. In this way, the replaced region was regarded irrelevant to the ground-truth class, and other regions were regarded relevant to the ground-truth class. Then, we **trained** an AlexNet (Krizhevsky et al., 2012) on the dataset by following (Yang & Kim, 2019), and computed the accurate Shapley value of the irrelevant region in the image. In this case, we only needed to compute sixteen Shapley values A_i^* for as few as sixteen image regions, whose computational cost was still affordable according to Equation (1) (if $N = 16$). Accurate Shapley values for all 16 image regions could be computed within ten minutes without any approximation.

As Figure 3 shows, the Shapley value of the irrelevant region was **not** zero. We also showed the relative attribution of the irrelevant region $\hat{k}$, $|A_{\hat{k}}^*|/\mathbb{E}_{i\in\Omega}[|A_i^*|]$, on the top of images, where Ω denotes the set of all regions in the image. The average relative attribution over all images was 1.0336. Therefore, it was inappropriate to assume that the irrelevant region has no attribution to the classification. Actually, Yilun et al. (2022); Jasmijn et al. (2021) proposed similar evaluation methods. We have discussed the problem with these studies in Appendix N.

Trustworthiness of using the ranked attributions for evaluation. Several previous studies (Samek et al., 2017; Ancona et al., 2018; Hooker et al., 2019; Kindermans et al., 2018b; Warnecke et al., 2020) proposed a novel idea, *i.e.,* using the ranked attribution values for evaluation. As Figure 4 (a) shows, for each time step t, they masked the input variable with the t-th lowest (or highest) attribution value in a sequential manner. In this way, if the attribution value was estimated more accurately, then the performance of the DNN was supposed to decrease more slowly (or more quickly).

However, Ribeiro et al. (2016); Chen et al. (2018); Yao et al. (2022) found that the measured numerical contribution of an input variable to the performance significantly depended on the order of input variables being masked. Beyond these experimental studies, in this paper, we propose Remark 1 based on findings in (Deng et al., 2021) to further clarify the unfairness of such ranking-based masking methods. Appendix H introduces more details of Remark 1. Such unfairness cannot be fully solved by the solution in (Yao et al., 2022).

Remark 1 *According to (Deng et al., 2021), attributions of different methods can be uniformly written into an allocation of independent effects and Taylor interaction effects, i.e.,* $a_i = \sum_{j\in\Omega}\phi_{j\to i} + \sum_{S\subseteq\Omega}\psi_{S\to i}$. *Here,* $\phi_{j\to i}$ *denotes a component of the independent effect of the variable* j*, which is allocated to the variable* i*'s attribution* a_i. $\psi_{S\to i}$ *denotes a component of the Taylor interaction effect between variables in* S*, which is allocated to the variable* i*'s attribution.* $\phi_{j\to i}$ *and* $\psi_{S\to i}$ *are all computed in (Deng et al., 2021), so that* $\psi_S = \sum_{i\in\Omega}\psi_{S\to i}$ *measures all interaction effects between variables in* S*. Thus, if we place the input variable* $i \in S$ *before all other variables in* S *in the masking order, then all interaction effects* ψ_S *will be all assigned with the variable* i*, without assigning any effects with other variables* $j \neq i$ *in* S*, i.e.,* $\forall i, j \in S, i \neq j, \psi_{S\to i} = \psi_S, \psi_{S\to j} = 0$.

Furthermore, we designed a new experiment to verify that an input variable's marginal attribution strongly depended on the order (or more precisely the context) when the variable was masked. If so,

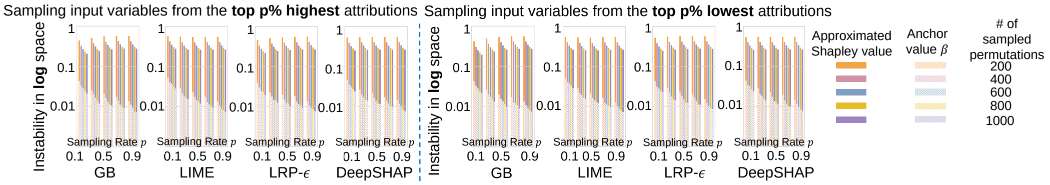

Figure 5: Instability of the approximated Shapley value and the instability of the anchor value $\beta = \mathbb{E}_{S:|S|=pN}[(\sum_{i\in S} A_i^{\text{shap}})/(|S|\|A^{\text{shap}}\|)]$ of our bias metric. The set of input variables in S were sampled according to attribution maps from different explanation methods. Results show that the anchor value of our metric exhibited a much lower instability than the approximated Shapley value.

the evaluation based on masking top-ranked input variables would not faithfully reflect true attributions of input variables. We trained a LeNet (LeCun et al., 1998) on the CIFAR-10 dataset (Krizhevsky & Hinton, 2009). Given a variable i in the input sample, we computed its marginal attribution when this variable was the $(t+1)$-th masked variable, as $\Delta Y(P_t, i) = Y_{\text{masking } P_t} - Y_{\text{masking } P_t \& i}$. P_t represents a set of t variables that had been masked before the t-th step. $Y_{\text{masking } P_t}$ denotes the network output when we fed the input sample with variables in P_t being masked into the DNN. Similarly, $Y_{\text{masking } P_t \& i}$ denotes the network output when we further masked an additional variable i. Then, $V_i = \mathbb{E}_t \mathbb{E}_{P_t:\|P_t\|=t}[\max(|\frac{\Delta Y(P_t,i)}{\Delta \bar{Y}(i)}|, |\frac{\Delta \bar{Y}(i)}{\Delta Y(P_t,i)}|)]$ measures the instability of $\Delta Y(P_t, i)$, when people masked the variable i at different orders with different contexts, where $\Delta \bar{Y}(i) = \mathbb{E}_{t'} \mathbb{E}_{P'_{t'}:\|P'_{t'}\|=t'}[\Delta Y(P'_{t'}, i)]$. Figure 4 (b) reports the distribution of $\Delta Y(P_t, i)$ over different orders and V_i values for specific variables. A large value of V_i indicates that the marginal attribution $\Delta Y(P_t, i)$ was unstable. We found that the marginal attribution of a variable was significantly affected by the order of masking. In particular, for some input variables, the marginal attribution $\Delta Y(P_t, i)$ was sometimes more than five times of the average marginal attribution of the same variable over different orders. Moreover, as Figure 4 (c) shows, the masked variables distributed over the whole input sample, and formed an irregular context without an explainable meaning. The evaluation under irregular unexplainable contexts might not reflect the true importance of the variable. Thus, it still had technical flaws to evaluate attribution methods based on the ranking order of attributions.

4 Experiments

To evaluate attribution methods, we conducted experiments on both tabular data and visual data for evaluation, which included a tabular dataset (the TV news channel commercial detection dataset (Vyas et al., 2014), namely the TV news dataset for short), the Pascal VOC 2012 (Everingham et al., 2010) dataset, and the CIFAR-10 (Krizhevsky & Hinton, 2009) dataset. In experiments, we evaluated the following explanation methods, including the Gradient (Grad) (Simonyan et al., 2013), Gradient×Input (GI) (Shrikumar et al., 2016), integrated gradient (IG) (Sundararajan et al., 2017), guided back-propagation (GB) (Springenberg et al., 2014), layer-wise relevance propagation (LRP) (Bach et al., 2015), DeepSHAP (Shrikumar et al., 2016), LIME (Ribeiro et al., 2016), and Grad-CAM (Selvaraju et al., 2017). Appendix I makes a survey on these attribution methods. Figure 1 shows attribution maps generated by these methods.

Implementation Details: To approximate the Shapley value for each pixel, we sampled the set T in Equation (1) for 1000 times for each pixel of images in the CIFAR-10 dataset, and sampled T for 100 times for each pixel of images in the Pascal VOC 2012 dataset. For the TV news dataset, we sampled T for 1000 times for each input variable. We sampled the top p (p=10%, 30%, 50%, 70%, 90%) testing input variables with the highest/lowest values. In this way, the proposed bias metric can consider all input variables for evaluation. Besides, LRP-ϵ was not used on residual networks, because the relevance propagation rules of some structures in ResNet were not defined, to the best of our knowledge.

Exp. 1: Stability of the proposed bias metric. This experiment compared the stability of the following two evaluation strategies, when they were conducted with the same computational cost. The strategies were 1. directly approximating relatively accurate attributions[1] for evaluation, and 2. our

[1] As introduced above Proposition 1, we applied (Castro et al., 2009) to approximate Shapley values, because other approximation methods usually generated biased results.

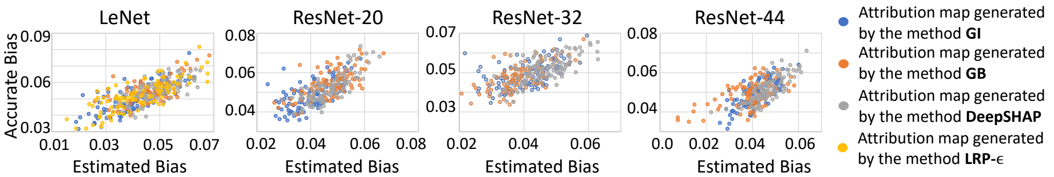

Figure 6: The positive relationship between the bias metric M_{bias} and the accurate bias (with an NP-hard computational cost). Each point corresponded to the two bias metrics computed on a specific attribution map. It verified that the bias metric M_{bias} was positively correlated with the accurate bias, which verified the objectiveness of the proposed bias metric.

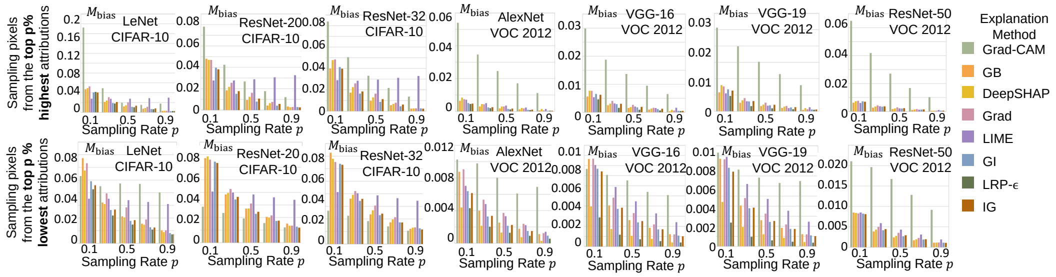

Figure 7: Bias of the attribution map at the level of input variables. LRP-ϵ provided attribution maps with the least bias on LeNet, AlexNet, and VGG-16/19. GI and GB outperformed other attribution methods on ResNets. Due to the limitation of the page number, we provide results on the ResNet-44/56/50 and explicit result numbers in Appendix K.

evaluation based on the bias of the attribution distribution, respectively. Given a trained DNN and an input sample $I \in \mathbf{I}$, we computed the approximated Shapley value multiple times. We normalized the approximated Shapley value as $\alpha = A^{\text{shap}}/\|A^{\text{shap}}\|$, just like Equation (3). We computed the anchor value for the proposed bias metric in Equation (3) $\beta = \mathbb{E}_{S:|S|=pN}[(\sum_{i\in S} A_i^{\text{shap}})/(|S|\|A^{\text{shap}}\|)]$ for multiple times. The instability of the approximated Shapley value was computed as $\text{Instability}_{\text{shap}} = \mathbb{E}_{I\in\mathbf{I}}\left[\mathbb{E}_{u,v:u\neq v}|\alpha_{(u)} - \alpha_{(v)}|/\mathbb{E}_w[|\alpha_{(w)}|]\right]$, and the instability of the anchor value of the bias metric was computed as $\text{Instability}_{\text{ours}} = \mathbb{E}_{I\in\mathbf{I}}\left[\mathbb{E}_{u,v:u\neq v}|\beta_{(u)} - \beta_{(v)}|/\mathbb{E}_w[|\beta_{(w)}|]\right]$. $\alpha_{(u)}$ and $\beta_{(u)}$ denoted the metrics of α and β computed at the u-th time, respectively.

In this experiment, we used the LeNet trained on the CIFAR-10 dataset. We explored the change of the instability along with the increase of the sampling number and the increase of the input variable number. As Figure 5 shows, the instability of our anchor value was much lower than the instability of the approximated Shapley value. This result demonstrated that the anchor value of the bias metric converged much faster, and thereby was more reliable than the approximated Shapley value.

Exp. 2: Trustworthiness of the proposed bias metric. It was a challenge to evaluate the trustworthiness of the proposed bias metric. We compared the proposed bias metric with the accurate bias, whose computation needed accurately computed Shapley values. Considering the exponential computational cost, we only computed relatively accurate Shapley values just for a few input variables using the method in (Castro et al., 2009) with a huge number of sampling. In this case, we could consider that the estimated Shapley values were accurate enough to verify the trustworthiness of the proposed bias metric. In this experiment, we used LeNet and ResNet-20/32/44 trained using the CIFAR-10 (Krizhevsky & Hinton, 2009) dataset. To this end, we sampled 10% input variables with the highest attributions to evaluate the proposed bias metric. Figure 6 compares the proposed bias metric with the accurate bias. Each point corresponded to a specific evaluation result. We found that the bias metric was roughly positively correlated with the accurate bias, which verified the objectiveness of the bias metric.

Exp. 3: Evaluating the bias of the attribution maps. Figure 7 and Figure 8 show bias values of attribution maps, which were generated by different attribution methods on tabular data and images, respectively. GI and GB provided the least biased attribution maps for ResNets. For AlexNet, VGG-16/19, and LeNet, LRP-ϵ outperformed other methods. Figure 8 shows that IG and DeepSHAP usually achieved the least biased attribution maps on the tabular classification task. Besides, we found that the performance of LIME was volatile, *i.e.,* in some cases, LIME performed quite well, but in some

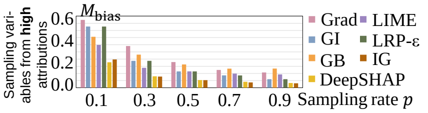

Figure 8: Bias of attribution maps estimated on the TV news channel commercial detection dataset (Vyas et al., 2014). IG and DeepSHAP outperformed other attribution methods.

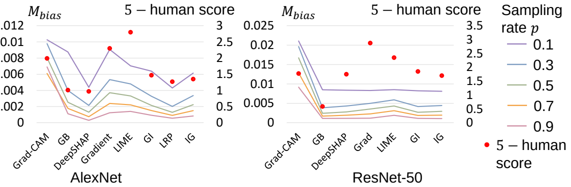

Figure 9: Comparison of the evaluation result based on our metric M_{bias} and that based on human intuition.

other cases, LIME performed the worst. This phenomenon was obvious in the CIFAR-10 dataset. It was because LIME calculated attribution maps for super-pixels. The number of super-pixels in an image from the CIFAR-10 dataset was limited. In this way, many pixels within a single super-pixel shared the same attribution value in results of LIME, which made it hard to sample these pixels with significantly biased attribution values.

In particular, we also found that our metric had no partiality to Shapley-value-based methods, such as DeepSHAP. Figure 2 shows the clear difference between the DeepSHAP's attribution map and relatively accurate Shapley values. We computed such relatively accurate Shapley values by sampling 20,000 testing samples from a single input image, to ensure its accuracy. It shows that the DeepSHAP's attribution map was dramatically different from relatively accurate Shapley values. Furthermore, (Aas et al., 2019) also showed that accurate Shapley values computed with an NP-hard cost were usually dramatically different from Shapley values approximated by practical methods. In Appendix L, we also conducted an experiment to show the unreliability of DeepSHAP.

Exp. 4: Sometimes conflicting with human intuition. Although our evaluation metric was designed to examine the objectiveness of attribution methods, we still conducted experiments to check whether the most objective attributions selected by our metric also fit human intuition. In this experiment, we used the AlexNet (Krizhevsky et al., 2012) and ResNet-50 (He et al., 2016) trained on the Pascal VOC 2012 dataset (Everingham et al., 2010) for analysis. We generated attribution maps based on different attribution methods. We had four human annotators to score these attribution maps from 1 to 5 based on their intuitive understandings of how much these attribution maps were consistent with their intuition. As Figure 9 shows, the metric M_{bias} of explaining the AlexNet well fit the annotated human rating, but M_{bias} of explaining the ResNet-50 had a large gap with the annotated human rating.

Seemingly contradictory with previous metrics, but actually not. We noticed that our evaluation results seemed to conflict with (Adebayo et al., 2018) and ROAR (Hooker et al., 2019), but actually not. Adebayo et al. (2018) criticized the GB for being insensitive to the randomization of DNNs. The evaluation metric ROAR (Hooker et al., 2019) showed that the removal of important input variables based on GB did not affect the performance of the DNN significantly. Theoretically, the evaluation in (Adebayo et al., 2018) naturally welcomed smoothed attributions (*e.g.* Grad-CAM), and criticized edge-like attributions (*e.g.* GB). However, this phenomenon also showed that edge-like pixels had large impacts on network features, which actually reflected the true phenomenon of signal processing in DNNs. Besides, ROAR might have a partiality to Grad-CAM. It was because removing pixels from a smooth surface/region was more likely to bring in additional noisy features than removing pixels from edges. To this end, GB usually assigned edges with large attributions, which caused the performance of the DNN was not significantly affected. In comparison, the proposed bias metric in this paper could measure the objectiveness of explanation methods more accurately than (Adebayo et al., 2018; Hooker et al., 2019). Please see Appendix J for more discussions.

5 Conclusion and Discussions

In this paper, we propose a metric, *i.e.,* the bias of attribution maps, to evaluate the objectiveness of attribution methods. The proposed evaluation metric is computed without a need for ground-truth explanations. We apply our metric to widely-used explanation methods. We have also verified the stability and trustworthiness of our metric in various experiments. Experimental results showed that our metric effectively evaluated the bias of attribution maps. Although the proposed metric makes a breakthrough in the evaluation of attribution methods, it still has a few limitations. Please see Appendix F for more discussions about limitations and potential social impacts of this paper.

REFERENCES

Kjersti Aas, Martin Jullum, and Anders Løland. Explaining individual predictions when features are dependent: More accurate approximations to shapley values. *arXiv:1903.10464*, 2019.

Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. Sanity checks for saliency maps. *In NIPS*, 2018.

David Alvarez-Melis and Tommi S. Jaakkola. Towards robust interpretability with self-explaining neural networks. *In arXiv:1806.07538*, 2018.

Marco Ancona, Enea Ceolini, Cengiz Oztireli, and Markus Gross. Towards better understanding of gradient-based attribution methods for deep neural networks. In *ICLR*, 2018.

Marco Ancona, Cengiz Oztireli, and Markus Gross. Explaining deep neural networks with a polynomial time algorithm for shapley values approximation. *arXiv:1903.10992*, 2019.

Leila Arras, Franziska Horn, Grégoire Montavon, Klaus-Robert Müller, and Wojciech Samek. "what is relevant in a text document?": An interpretable machine learning approach. *PLoS ONE*, 12: E0181142, 08 2017. doi: 10.1371/journal.pone.0181142.

Leila Arras, Ahmed Osman, Klaus-Robert Müller, and Wojciech Samek. Evaluating recurrent neural network explanations. *In arXiv:1904.11829*, 2019.

Sebastian Bach, Alexander Binder, Gregoire Montavon, Frederick Klauschen, Klausrobert Muller, and Wojciech Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PLOS ONE*, 10(7), 2015.

David Bau, Bolei Zhou, Aditya Khosla, Aude Oliva, and Antonio Torralba. Network dissection: Quantifying interpretability of deep visual representations. *In CVPR*, 2017.

Umang Bhatt, Adrian Weller, and José M. F. Moura. Evaluating and aggregating feature-based model explanations. *IJCAI*, 2020.

Alexander Binder, Grégoire Montavon, Sebastian Lapuschkin, Klaus-Robert Müller, and Wojciech Samek. Layer-wise relevance propagation for neural networks with local renormalization layers. *In International Conference on Artificial Neural Networks (ICANN)*, 2016.

Oana-Maria Camburu, Eleonora Giunchiglia, Jakob Foerster, Thomas Lukasiewicz, and Phil Blunsom. Can i trust the explainer? verifying post-hoc explanatory methods. *arXiv: 1910.02065*, 2019.

Javier Castro, Daniel Gómez, and Juan Tejada. Polynomial calculation of the shapley value based on sampling. *In Computers & Operations Research*, 36(5):1726–1730, 2009.

Jianbo Chen, Le Song, Martin J Wainwright, and Michael I Jordan. L-shapley and c-shapley: Efficient model interpretation for structured data. *arXiv preprint arXiv:1808.02610*, 2018.

Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. *In NIPS*, 2016.

Ian Covert and Su-In Lee. Improving kernelshap: Practical shapley value estimation using linear regression. In *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, pp. 3457–3465, 2021.

Xiaocong Cui, Jung Min Lee, and J Hsieh. An integrative 3c evaluation framework for explainable artificial intelligence. *In The annual Americas Conference on Information Systems (AMCIS)*, 2019.

Huiqi Deng, Na Zou, Mengnan Du, Weifu Chen, Guocan Feng, and Xia Hu. Understanding and Unifying Fourteen Attribution Methods with Taylor Interactions. *arXiv preprint arXiv:2105.13841*, 2021.

Jay DeYoung, Sarthak Jain, Nazneen Rajani, Eric P. Lehman, Caiming Xiong, Richard Socher, and Byron C. Wallace. Eraser: A benchmark to evaluate rationalized nlp models. *ArXiv*, abs/1911.03429, 2020.

Alexey Dosovitskiy and Thomas Brox. Inverting visual representations with convolutional networks. *In CVPR*, 2016.

M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *In International Journal of Computer Vision*, 88(2):303–338, June 2010.

Ruth Fong and Andrea Vedaldi. Net2vec: Quantifying and explaining how concepts are encoded by filters in deep neural networks. *In CVPR*, 2018.

Ruth C. Fong and Andrea Vedaldi. Interpretable explanations of black boxes by meaningful perturbation. *In arXiv:1704.03296v1*, 2017.

Christopher Frye, Damien de Mijolla, Tom Begley, Laurence Cowton, Megan Stanley, and Ilya Feige. Shapley explainability on the data manifold. In *ICLR*, 2021.

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

Emil Julius Gumbel. Les valeurs extrêmes des distributions statistiques. In *Annales de l'institut Henri Poincaré*, volume 5, pp. 115–158, 1935.

John C Harsanyi. A simplified bargaining model for the n-person cooperative game. *International Economic Review*, 4(2):194–220, 1963.

Peter Hase, Harry Xie, and Mohit Bansal. The out-of-distribution problem in explainability and search methods for feature importance explanations. In *NeurIPS*, 2021.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *In CVPR*, 2016.

Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. β-vae: learning basic visual concepts with a constrained variational framework. *In ICLR*, 2017.

Sara Hooker, Dumitru Erhan, Pieterjan Kindermans, and Been Kim. Evaluating feature importance estimates. In *NIPS*, 2019.

Zhiting Hu, Xuezhe Ma, Zhengzhong Liu, Eduard Hovy, and Eric P. Xing. Harnessing deep neural networks with logic rules. *In arXiv:1603.06318v2*, 2016.

Georgin Jacob, RT Pramod, Harish Katti, and SP Arun. Qualitative similarities and differences in visual object representations between brains and deep networks. *Nature communications*, 12(1): 1872, March 2021. ISSN 2041-1723. doi: 10.1038/s41467-021-22078-3.

Bastings Jasmijn, Ebert Sebastian, Zablotskaia Polina, Sandholm Anders, and Filippova Katja. 'will you find these shortcuts?' a protocol for evaluating the faithfulness of input salience methods for text classification. In *ArXiv:2111.07367*, 2021.

Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, and Rory Sayres. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). *In ICML*, 2018.

Pieter-Jan Kindermans, Kristof T. Schütt, Maximilian Alber, Klaus-Robert Müller, Dumitru Erhan, Been Kim, and Sven Dähne. Learning how to explain neural networks: Patternnet and patternattribution. *In ICLR*, 2018a.

Pieterjan Kindermans, Kristof T Schutt, Maximilian Alber, Klausrobert Muller, Dumitru Erhan, Been Kim, and Sven Dahne. Learning how to explain neural networks: Patternnet and patternattribution. In *ICLR*, 2018b.

PangWei Koh and Percy Liang. Understanding black-box predictions via influence functions. *In ICML*, 2017.

A. Krizhevsky and G. Hinton. Learning multiple layers of features from tiny images. *In Computer Science Department, University of Toronto, Tech. Rep*, 1, 2009.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. *In NIPS*, 2012.

I. Elizabeth Kumar, Suresh Venkatasubramanian, Carlos Scheidegger, and Sorelle A. Friedler. Problems with shapley-value-based explanations as feature importance measures. In *ICML 2020*, 2020.

Yann LeCun, Lèon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *In Proceedings of the IEEE*, 1998.

Renjie Liao, Alex Schwing, Richard Zemel, and Raquel Urtasun. Learning deep parsimonious representations. *In NIPS*, 2016.

Scott M. Lundberg and Su-In Lee. A unified approach to interpreting model predictions. *In NIPS*, 2017.

Aravindh Mahendran and Andrea Vedaldi. Understanding deep image representations by inverting them. *In CVPR*, 2015.

Dong Nguyen. Comparing automatic and human evaluation of local explanations for text classification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pp. 1069–1078, 2018.

Jose Oramas, Kaili Wang, and Tinne Tuytelaars. Visual explanation by interpretation: Improving visual feedback capabilities of deep neural networks. *ICLR*, 2019.

Arijit Ray, Michael Cogswell, Xiaoyu Lin, Kamran Alipour, Ajay Divakaran, Yi Yao, and Giedrius Burachas. Generating and evaluating explanations of attended and error-inducing input regions for vqa models. *arXiv:2103.14712*, 2021.

Jie Ren, Mingjie Li, Qirui Chen, Huiqi Deng, and Quanshi Zhang. Towards axiomatic, hierarchical, and symbolic explanation for deep models. In *ArXiv:2111.06206*, 2021.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should i trust you?" explaining the predictions of any classifier. *In KDD*, 2016.

Tomsett Richard, Harborne Dan, Gurram Supriyo, Chakraborty annd Prudhvi, and Preece Alun. Sanity check for saliency metrics. In *AAAI*, 2020.

Sara Sabour, Nicholas Frosst, and Geoffrey E. Hinton. Dynamic routing between capsules. *In NIPS*, 2017.

Wojciech Samek, Alexander Binder, Gregoire Montavon, Sebastian Lapuschkin, and Klausrobert Muller. Evaluating the visualization of what a deep neural network has learned. *IEEE Transactions on Neural Networks*, 28(11):2660–2673, 2017.

Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. *In ICCV*, 2017.

Lloyd S Shapley. A value for n-person games. *In Contributions to the Theory of Games*, 2(28): 307–317, 1953.

Avanti Shrikumar, Peyton Greenside, Anna Shcherbina, and Anshul Kundaje. Not just a black box: Learning important features through propagating activation differences. *In arXiv:1605.01713*, 2016.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *In ICLR*, 2015.

Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *In arXiv:1312.6034*, 2013.

Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. Striving for simplicity: The all convolutional net. *In arXiv:1412.6806*, 2014.

Austin Stone, Huayan Wang, Yi Liu, D. Scott Phoenix, and Dileep George. Teaching compositionality to cnns. *In CVPR*, 2017.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *International Conference on Machine Learning*, pp. 3319–3328. PMLR, 2017.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *In arXiv:1312.6199v4*, 2014.

Minh N Vu, Truc D Nguyen, NhatHai Phan, Ralucca Gera, and My T Thai. Evaluating explainers via perturbation. *In arXiv:1906.02032*, 2019.

Apoorv Vyas, Raghvendra Kannao, Vineet Bhargava, and Prithwijit Guha. Commercial block detection in broadcast news videos. In *ICVGIP '14*, 2014.

Guanchu Wang, Yu-Neng Chuang, Mengnan Du, Fan Yang, Quan Zhou, Pushkar Tripathi, Xuanting Cai, and Xia Hu. Accelerating shapley explanation via contributive cooperator selection. In *Proceedings of the 39th International Conference on Machine Learning*, pp. 22576–22590, 2022.

Alexander Warnecke, Daniel Arp, Christian Wressnegger, and Konrad Rieck. Evaluating explanation methods for deep learning in security. *IEEE European Symposium on Security and Privacy*, 2020.

Tianfu Wu, Xilai Li, Xi Song, Wei Sun, Liang Dong, and Bo Li. Interpretable r-cnn. *In arXiv:1711.05226*, 2017.

Fan Yang, Mengnan Du, and Xia Hu. Evaluating explanation without ground truth in interpretable machine learning. *In arxiv: 1907.06831*, 2019.

Mengjiao Yang and Been Kim. Bim: Towards quantitative evaluation of interpretability methods with ground truth. *In arXiv:1907.09701*, 2019.

Rong Yao, Leemann Tobias, Borisov Vadim, Kasneci Gjergji, and Kasneci Enkelejda. A consistent and efficient evaluation strategy for attribution methods. In *ICML*, 2022.

Chihkuan Yeh, Chengyu Hsieh, Arun Sai Suggala, David I Inouye, and Pradeep Ravikumar. On the (in)fidelity and sensitivity of explanations. In *NeurIPS*, pp. 10965–10976, 2019.

Zhou Yilun, Booth Serena, Riberio Marco Tulio, and Shah Julie. Do feature attribution methods correctly attribute features? In *AAAI*, 2022.

Jason Yosinski, Jeff Clune, Anh Nguyen, Thomas Fuchs, and Hod Lipson. Understanding neural networks through deep visualization. *In ICML Deep Learning Workshop*, 2015.

Matthew D. Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. *In ECCV*, 2014.

Quanshi Zhang, Ying Nian Wu, and Song-Chun Zhu. Interpretable convolutional neural networks. *In CVPR*, 2018.

Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. *In CVPR*, 2016.

A More examples of attribution maps

In this section, we provide more attribution maps generated by different explanation methods, which is shown in Figure 10.

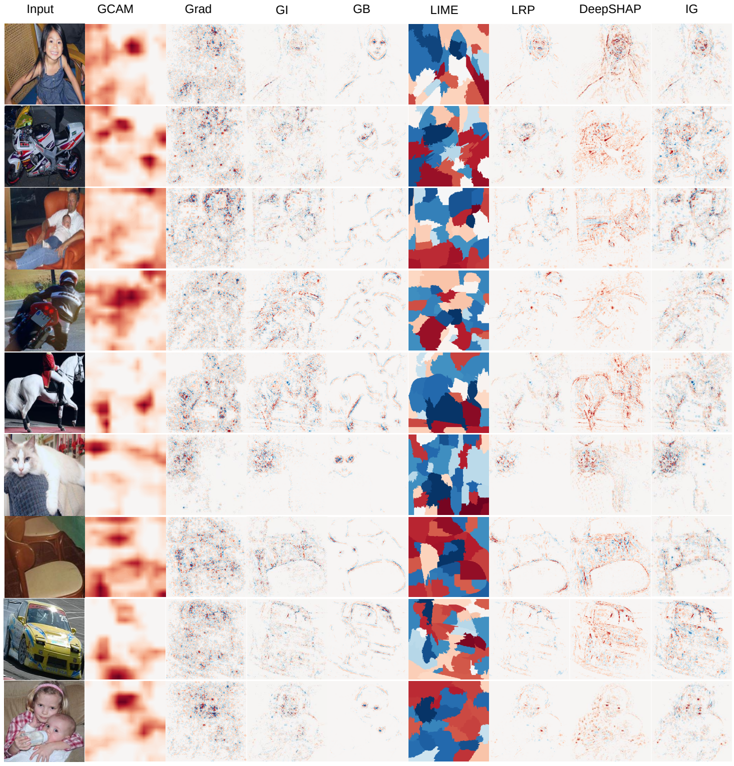

Figure 10: Examples of attribution maps.

B Related Work

Section 3 has discussed previous studies that attempted on the evaluation of attribution methods. In this section, we continue to discuss more about technical details of attribution methods, although such techniques are not directly related to the evaluation. Nevertheless, we will move this section to the main paper when the paper is accepted.

For the evaluation of attribution/importance/saliency explanation methods, the qualitative analysis of explanation results via subject judgement (Cui et al., 2019; Yang et al., 2019; Nguyen, 2018) is a classical perspective for the evaluation. However, we would like to limit our discussions to the quantitative evaluation metric for explanation methods.

Yang & Kim (2019) and Oramas et al. (2019) built a specific dataset to generate intuitive ground-truth explanations for evaluation. However, in Section 3, we find that we cannot assume the DNN not to use noises for classification, which hurts the trustworthiness of the evaluation. Another kind of classical evaluation metrics (Samek et al., 2017; Hooker et al., 2019; Ancona et al., 2018; Fong & Vedaldi, 2017; Kindermans et al., 2018b; Warnecke et al., 2020; DeYoung et al., 2020; Hase et al., 2021; Arras et al., 2017; Nguyen, 2018) was sequentially removing pixels from pixels with the lowest attributions to those with the highest attributions (or from the highest attributions to the lowest attributions). They used the decreasing speed of the DNN's performance to evaluate the quality of explanation results. If people masked pixels with the lowest attribution values first, then a slow decrease of the network output score indicated a high quality of the attribution map. However, previous studies (Richard et al., 2020; Yao et al., 2022) have pointed out that the method of sequentially removing pixels could not objectively reflect the true importance of pixels. Section 3 has discussed this issue both theoretically and experimentally.

Besides, some studies evaluated explanation results from other perspectives. Arras et al. (2019) and Vu et al. (2019) evaluated attribution maps from the perspective of adversarial attacks by adding random noise to the input. Some studies (Adebayo et al., 2018; Ghorbani et al., 2019; Alvarez-Melis & Jaakkola, 2018) evaluated the robustness of explanation methods *w.r.t.* the perturbation. Adebayo et al. (2018) randomized layers of a DNN from the top to the bottom, and visualized the change of attribution maps. Bhatt et al. (2020) proposed three desirable properties of the explanation methods, including low sensitivity, high faithfulness, and low complexity to evaluate explanation methods. Warnecke et al. (2020) used multiple perspectives to evaluate explanations. Yeh et al. (2019) evaluated explanation methods using (in)fidelity and sensitivity. Table 2 summarizes various evaluation perspectives of previous studies.

Table 2: Comparisons of perspectives of evaluating explanation results among different studies.

Evaluation metrics	Perspective
Yeh et al. (2019), Yang et al. (2019)	(In)fidelity
Adebayo et al. (2018), Yeh et al. (2019) Bhatt et al. (2020)	Sensitivity
Warnecke et al. (2020), Bhatt et al. (2020)	Sparsity
Ghorbani et al. (2019), Warnecke et al. (2020) Alvarez-Melis & Jaakkola (2018)	Robustness
Our metric, Arras et al. (2019) Yang & Kim (2019), Samek et al. (2017)	Objectiveness

Unlike previous studies, in this paper, we focus on evaluating the objectiveness of explanation methods. We believe that the evaluation of the objectiveness is most important to explanation methods. In comparison, the infidelity mainly reflects the non-linearity of the explained model, instead of evaluating the trustworthiness of explanation methods. As for the sensitivity, if the DNN is sensitive to random noises, then an object explanation result is also supposed to be sensitive to such noises. Similarly, if the DNN is not robust to the adversarial attack, then an objective explanation result is supposed not to be robust to the same attack. The sparsity of an explanation method cannot reflect the objectiveness of this explanation method, *i.e.* a sparse explanation method may be not objective enough, and an objective explanation method may be not sparse.

Other kinds of explanation methods. In this paper, we mainly focus on the evaluation of attribution/importance/saliency explanation methods. However, there are many other kinds of explanation methods.

Firstly, the visualization of feature representations inside a DNN was the most direct way of opening the black-box of the DNN (Zeiler & Fergus, 2014; Mahendran & Vedaldi, 2015; Yosinski et al., 2015; Dosovitskiy & Brox, 2016). Secondly, other studies diagnosed feature representations inside a DNN (Kindermans et al., 2018a; Koh & Liang, 2017; Szegedy et al., 2014; Bau et al., 2017; Fong & Vedaldi, 2018). Thirdly, a recent new trend was to learn interpretable features in DNNs (Hu et al., 2016; Stone et al., 2017; Liao et al., 2016; Chen et al., 2016; Higgins et al., 2017). Capsule nets (Sabour et al., 2017) and the interpretable RCNN (Wu et al., 2017) learned interpretable features in intermediate layers.

Moreover, Kim et al. (2018) proposed a method to discover visual concepts encoded by the DNN, and computed attributions for the discovered concepts. However, the method (Kim et al., 2018) could only extract the attribution for each mid-level concept, instead of extracting pixel-wise or regional attributions. For example, it was difficult to compute the exact receptive field (pixels) corresponding to a specific "color" concept, so the method in (Kim et al., 2018) usually could not directly estimate attributions for an exact region/pixel. Therefore, we could not apply the evaluation metric toward pixel-wise/regional attributions to evaluate (Kim et al., 2018).

C Analysis of our metric and previous desirable axioms

In the "Axioms for objectiveness" part of Section 2, we have introduced several desirable axioms, which were proposed by previous studies, for the evaluation. Besides, we mentioned that the proposed metric potentially replects these axioms in the third paragraph of page 2. In this section, we will discuss how to derive our metric from previous desirable axioms. As aforementioned, previous studies proposed various desirable axioms for the objectiveness of explanation methods, including the *linearity axiom*, the *dummy axiom*, the *symmetry axiom*, the *efficiency axiom* (Shapley, 1953), and the *interaction distribution axiom* (Deng et al., 2021). Previous studies (Shapley, 1953; Deng et al., 2021) also proved that the Shapley value was the unique unbiased attribution method that satisfied the above five axioms.

However, the Shapley value cannot be directly used for the evaluation due to the unaffordable computational cost. To this end, we propose an evaluation metric using the same theoretical foundation as the Shapley value, but with a lower computational cost than the accurate Shapley value. Since our metric is proposed on the basis of the Shapley value, it can be considered to potentially satisfy previous desirable axioms.

D Can we directly use the proposed metric to estimate pixel-wise attributions?

In the first paragraph of page 5, we have mentioned that we do not use the proposed metric for explanations. In this paragraph, we will further discuss this issue. In this paper, the proposed metric is is **NEITHER** an approximation of the Shapley value, **NOR** a technique that can potentially be used to approximate the Shapley value. Indeed, this metric is designed to represent the bias of the entire attribution distribution over all pixels. For example, in the second row and the second column of Fig. 4(c) in the paper, pixels with the lowest 25% attributions almost uniformly and randomly distribute over the dress, the baggage, two arms, and the background. Our metric computes the average attribution using these pixels. Obviously, our metric is not an approximation of the attribution value for a specific pixel. Thus, the proposed metric just represents the bias of the entire attribution distribution over all pixels, instead of measuring the attribution for a specific pixel. Furthermore, people cannot infer the pixel-wised attribution value from the metric.

E Details about axioms of the Shapley value.

In Section 2, we have introduced several desirable axioms for the evaluation. In this section, we provide more details about these axioms. We continue using the notation in Section 2 to introduce these desirable axioms. As aforementioned in the "Axioms for objectiveness" part of Section 2 in the paper, an objective explanation method is supposed to satisfy the following axioms.
Linearity axiom: If the reward of a game F satisfies $F(T) = G(T) + H(T)$, where G and H are another two games. Then the Shapley value of each player $i \in \Omega$ in the game F is the sum of Shapley values of the player i in the game G and H, *i.e.* $A_i^{(F)} = A_i^{(G)} + A_i^{(H)}$.
Dummy axiom: The dummy player is defined as the player that satisfies $\forall T \subseteq \Omega \backslash \{i\}, F(T \cup \{i\}) = F(T) + F(\{i\})$. The dummy player i satisfies $F(\{i\}) - F(\emptyset) = A_i$, *i.e.* the dummy player has no interaction with any other players in Ω.
Symmetry axiom: If $\forall T \subseteq \Omega \setminus \{i, j\}, F(T \cup \{i\}) = F(T \cup \{j\})$, then $A_i = A_j$.
Efficiency axiom: $\sum_{i \in \Omega} F_i = F(\Omega) - F(\emptyset)$. The efficiency property ensures the overall reward can be distributed to each player in the game.
Interaction distribution axiom: $A_i = \sum_{T \subseteq \Omega : i \in T} D(T)/|T|$, where $D(T)$ represents the Harsanyi dividend (Harsanyi, 1963) of the set of players T. The Harsanyi dividend $D(T)$ measures the numerical utility created by the interaction patterns among exactly all players in T.

F Limitations and potential social impacts of this paper

About the limitation of this paper. In this paper, our evaluation metric mainly focuses on explanation methods that extract attributions. In fact, many people use heatmaps to explain a DNN's regional representation on an image. Some of heatmaps represent attributions/importances of input variables, but other heatmaps do not represent attributions (Ray et al., 2021). Our evaluation metric can evaluate heatmaps that measure attributions/importances of input variables, cannot evaluate heatmaps do not represent attributions. Another example is the method proposed by (Ray et al., 2021), which has been discussed in Appendix B. Kim et al. (2018) proposed a method to discover visual concepts encoded by the DNN, such as a specific "color" concept. In this case, we cannot directly apply out metric to evaluate (Kim et al., 2018).

Another limitation of this paper comes from technical flaws of the Shapley value. Kumar et al. (2020) pointed out two technical flaws of the Shapley value. First, the Shapley value could not totally avoid the OOD problem with the setting of baseline values. Second, the selection of players (*i.e.,* the partition of input variables) also affected the Shapley value. They are both open problems for Shapley values.

As for the first problem, the OOD problem is still ill-defined currently on strong heuristic assumptions. In this paper, we aim to evaluate the objectiveness of attribution maps in the scenario where the baseline value has been given. In this way, the OOD problem is orthogonal to the objectiveness of the attribution, under the condition that the baseline value has been given. Therefore, the OOD problem does not affect the trustworthiness of the proposed evaluation metric.

Similarly, as for the second problem, the target of this study is to evaluate the objectiveness of attribution maps under the assumption that the partition of input variables has been given. Thus, our attribution objectiveness problem is actually orthogonal to the problem in (Kumar et al., 2020), and these problems do not hurt the soundness of our study.

About the social impact of this paper. As for the potential social impact, this paper proposed an evaluation metric to evaluate existing attribution methods, which may solve the problem that people usually cannot obtain ground-truth explanations of a black-box model to evaluate the explanation for the model. Therefore, the negative social impact of this paper is negligible, and this study mainly has positive social impact.

G The approximation of the Shapley value and the analysis of the computational cost

In Proposition 1 we briefly clarify the computational cost of the approximation of the Shapley value. In this section, we will introduce the algorithm to approximate the Shapley value in detail, and prove that the variance of our metric is much lower than the variance of the approximated Shapley value.

Approximation of Shapley value. The Shapley value (Shapley, 1953) was proposed to compute the attribution distribution over all players in a cooperative game. However, it is an NP-hard problem to compute the accurate Shapley value. To this end, in this paper, we apply the sampling-based algorithm proposed by (Castro et al., 2009) to approximate the Shapley value. We continue using the notation in the "Properties of the Shapley value" section of the supplementary material to introduce the algorithm and analyze the computational cost. According to (Castro et al., 2009), the Shapley value can be computed by enumerating all possible permutations of pixels. Let O denote a permutation of pixels, and $\pi(\Omega)$ denotes the set of all possible permutations. For each pixel i, we use $\text{Pre}_i(O)$ to present the set of pixels in front of i in the permutation $O \in \pi(\Omega)$. In this way, the Shapley value of the pixel i can be computed as

$$A_i^{\text{shap},*} = \frac{1}{N!} \sum_{O \in \pi(\Omega)} [F(\text{Pre}_i(O) \cup \{i\}) - F(\text{Pre}_i(O))]$$

where N is the number of all pixels, and $F(\text{Pre}_i(O))$ is computed by remaining the pixels in $\text{Pre}_i(O)$ and replacing the pixels in $\Omega \setminus \text{Pre}_i(O)$ with the reference value. Note that the setting of reference values without the out-of-distribution problem is also important for the computation of Shapley values, but it is still an open problem (Ancona et al., 2019; Lundberg & Lee, 2017; Frye et al., 2021). Thus, in this paper, we follow the most widely used setting (*i.e.* setting the reference value as the average pixel value over images) (Ancona et al., 2019). Nevertheless the computation of Shapley value can also be conducted with other settings of reference values, which ensures the broad applicability. Besides, note that different permutations have the same weight to compute the average value.

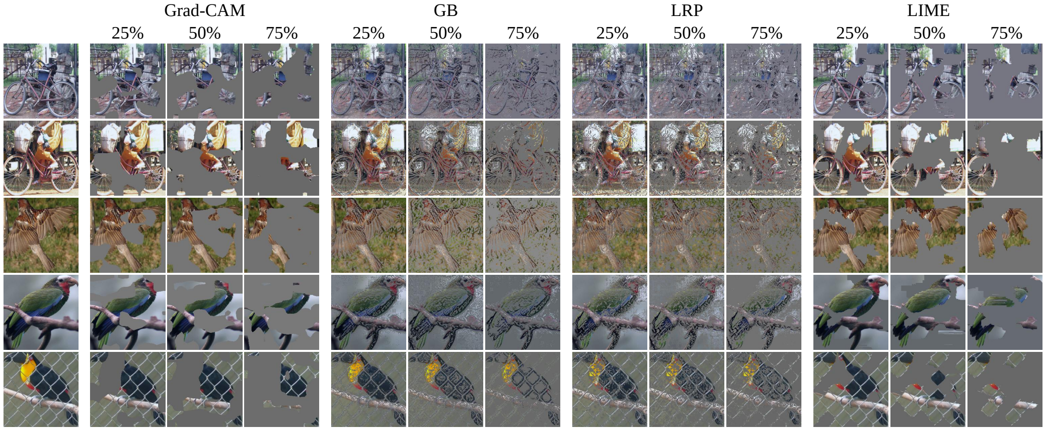

Figure 11: Visualization of images during the removal of pixels with top-ranked attributions. We gradually remove top-ranked 25%, 50%, and 75% pixels with the highest attribution values. Grad-CAM and LIME usually remove pixels from a region. GB and LRP usually first remove edge pixels in the image.

Based on the above equation, Castro et al. (2009) sampled permutations m times from $\pi(\Omega)$. Then the Shapley value of i can be approximated as follows.

$$\begin{aligned} A_i^{\text{shap}} &= \frac{1}{m^{\text{shap}}} \sum_{k=1}^{m^{\text{shap}}} \left[F(\text{Pre}_i(O_k) \cup \{i\}) - F(\text{Pre}_i(O_k)) \right], \\ \text{s.t.} \quad & \forall k,\, O_k \in \pi(\Omega) \end{aligned}$$

Computational cost of approximating Shapley value. Note that for one permutation, we can compute the marginal contribution $F(\text{Pre}_i(O_k) \cup \{i\}) - F(\text{Pre}_i(O_k))$ for all pixels by running the DNN $(N+1)$ times. Therefore, we can approximate the Shapley value A_i^{shap} for all N pixels using these m^{shap} permutations, and the computational cost is $\mathcal{O}(m^{\text{shap}}N)$.

Computational cost of our evaluation metric. Although the above method can approximate the Shapley value with a polynomial computational cost, this method could not provide accurate enough results for evaluation. To this end, instead of directly computing the Shapley value, we design a new metric with high accuracy but a low computational cost to estimate the attribution bias for evaluation.

Specifically, we can prove that the computational cost of our metric is much lower than the approximated Shapley value obtained by the above sampling. Suppose that we sample m times to compute the anchor value of our metric. According to (Castro et al., 2009), the variance of A_i^{shap} is σ^2/m where σ^2 satisfies

$$\sigma^2 = \sum_{O \in \pi(\Omega)} \frac{1}{N!} \left[F(\text{Pre}_i(O) \cup \{i\}) - F(\text{Pre}_i(O)) - A_i^{\text{shap},*} \right]^2$$

Let $(\sigma^{\text{shap}})^2$ denote the variance of A_i^{shap}, and we have $(\sigma^{\text{shap}})^2 = \sigma^2/m$. For the set of the sampled pixels S, the variance of the anchor value in our metric is $\frac{|S|(\sigma^{\text{shap}})^2}{|S|^2} = \frac{(\sigma^{\text{shap}})^2}{|S|} = \frac{\sigma^2}{m|S|}$, where m is the number of sampling for the computation of the anchor value in our metric. Apparently, if we want the approximated Shapley value A_i^{shap} to get the same stability as the anchor value in our metric, we need to sample $m^{\text{shap}} = |S|m$ times, which needs $|S|$ times computational cost as our metric.

H A DETAILED INTRODUCTION OF UNIFIED ATTRIBUTION FRAMEWORK IN (DENG ET AL., 2021)

In Remark 1, we brifly introduce the unified framework proposed by (Deng et al., 2021), and how this framework clarifies the unfariness of ranking-based masking evaluation methods. In this section, we will provide more introduction on this unified attribution framework.

Theorem 2 *Given a pre-trained DNN F and a given input sample $\boldsymbol{x} = [x_1, \ldots, x_n]^T \in \mathbb{R}^n$ indexed by $\Omega = [1, \ldots, n]$, the attribution a_i of the input variable x_i estimated by different methods can be uniformly written into an allocation of Taylor independent effect and Taylor interaction effects, i.e., $a_i = \sum_{j\in\Omega} \phi_{j\to i} + \sum_{S\subseteq\Omega} \psi_{S\to i}$.*

Taylor expansion of a pre-trained DNN. Given a pre-trained DNN F and a given input sample $\boldsymbol{x} = [x_1, \ldots, x_n]^T \in \mathbb{R}^n$ indexed by $\Omega = [1, \ldots, n]$, let us consider the K-order Taylor expansion of the DNN F, which is expanded at a baseline point $\boldsymbol{b} = [b_1, \ldots, b_n]$.

$$\begin{aligned} F(\boldsymbol{x}) &= F(\boldsymbol{b}) + \sum_{i=1}^{n} \frac{1}{1!} \frac{\partial F(\boldsymbol{b})}{\partial x_i}(x_i - b_i) + \sum_{i=1}^{n}\sum_{j=1}^{n} \frac{1}{2!} \frac{\partial^2 F(\boldsymbol{b})}{\partial x_i \partial x_j}(x_i - b_i)(x_j - b_j) + \cdots \\ &= F(\boldsymbol{b}) + \sum_{k=1}^{K} \sum_{\boldsymbol{\kappa}\in O_k} \underbrace{[\frac{1}{k!}\binom{k}{\kappa_1,\cdots,\kappa_n} \cdot \frac{\partial^k F(\boldsymbol{b})}{\partial^{\kappa_1} x_1 \cdots \partial^{\kappa_n} x_n}] \cdot \prod_{i=1}^{n}(x_i - b_i)^{\kappa_i}}_{\text{denoted by} I(\boldsymbol{\kappa})} \end{aligned} \tag{4}$$

In the above equation, the DNN $F(\boldsymbol{x})$ is decomposed into expansion terms of different orders, and each k-th order has numerous expansion terms in the set of O_k. Let $I(\boldsymbol{\kappa})$ denote an expansion term with the degree vector $\boldsymbol{\kappa} = [\kappa_1, \cdots, \kappa_n] \in O_k$, where $O_k = \{\boldsymbol{\kappa} | \kappa_i \in \mathbb{N}$ and $\sum_{i=1}^{n} \kappa_i = k\}$ is a set of all above degree vectors, corresponding to all expansion terms of the k-th order.

Taylor independent effect and Taylor interaction effect. All Taylor expansion terms $I(\boldsymbol{\kappa})$ of the DNN can be divided into two types of effects, (i) *Taylor interaction effect* $I(\boldsymbol{\kappa})$ where $|\boldsymbol{\kappa}|_0 > 1$, which represents a collaboration relationship between multiple input variables. (ii) *Taylor independent effect* $I(\boldsymbol{\kappa})$ where $|\boldsymbol{\kappa}|_0 = 1$, which is caused by a single input variable working independently. To avoid ambiguity, we use $\psi(\boldsymbol{\kappa})$ and $\phi(\boldsymbol{\kappa})$ to denote the Taylor interaction effect and the Taylor independent effect, respectively.

Unifying attribution methods by Taylor independent effect and interaction effect. Furthermore, Deng et al. (2021) have proven that attributions of different methods can be uniformly written into an allocation of Taylor independent effects and Taylor interaction effects, *i.e.*, the attribution a_i of the input variable x_i all can be reformulated as follows,

$$a_i = \underbrace{\sum_{j\in\Omega} \sum_{\boldsymbol{\kappa}\in Q_j} w_{i,\boldsymbol{\kappa}} \phi(\boldsymbol{\kappa})}_{\text{denoted by } \phi_{j\to i}} + \underbrace{\sum_{S\subseteq\Omega} \sum_{\boldsymbol{\kappa}\in Q_S} w_{i,\boldsymbol{\kappa}} \psi(\boldsymbol{\kappa})}_{\text{denoted by } \psi_{S\to i}} = \sum_{j\in\Omega} \phi_{j\to i} + \sum_{S\subseteq\Omega} \psi_{S\to i} \tag{5}$$

where $Q_j = \{\boldsymbol{\kappa} | \kappa_j > 0; \forall i \neq j, \kappa_i = 0\}$ and $Q_S = \{\boldsymbol{\kappa} | \forall i \in S, \kappa_i > 0; \forall i \notin S, \kappa_i = 0\}$.

I EXPERIMENTAL DETAILS

In the first two paragraphs of Section 4, we briefly introduced our experimental settings and the evaluated explanation methods in this paper. In this section, we provided more details about our experiments.

We conducted experiments using the Pascal VOC 2012 (Everingham et al., 2010) dataset, the CIFAR-10 (Krizhevsky & Hinton, 2009) dataset, and the TV news channel commercial detection dataset (a tabular dataset) (Vyas et al., 2014). The Pascal VOC 2012 dataset is mainly used for object detection. Just like (Zhang et al., 2018), we cropped objects using their bounding boxes. We used the cropped objects as inputs to train DNNs for multi-category classification. AlexNet (Krizhevsky et al., 2012), VGG-16/19 (Simonyan & Zisserman, 2015), ResNet-50/101 (He et al., 2016) were trained using the Pascal VOC 2012 dataset. We trained and explained LeNet (LeCun et al., 1998), ResNet-20/32/44/56 (He et al., 2016) using the CIFAR-10 dataset. As for the tabular dataset, we trained an MLP-5 using the TV news channel commercial detection dataset.

In this paper, we evaluated the following explanation methods.
Grad: Given an input, Simonyan et al. (2013) quantified the attribution value with the gradient of the input. We termed this algorithm as Grad. For RGB images with multiple channels, Grad selected the maximum magnitude across all channels for each pixel.

GI: Shrikumar et al. (2016) proposed a method, namely Gradient×Input, which used the pixel-wise product of the input and its gradient as the attribution value. Attribution values for RGB channels were summed up to get the final attribution value.
IG: Integrated Gradient, namely IG, was proposed by (Sundararajan et al., 2017), which integrated the gradient along the straight path from an empty input to the target input.
GB: Guided Back-propagation, namely GB, corresponded to Grad, where the back-propagation rule at ReLU units was redefined (Springenberg et al., 2014).
LRP-ϵ: Layer-wise relevance propagation (LRP) (Bach et al., 2015) redefined back-propagation rules for each layer to decompose the output of a DNN over the input. We used LRP-ϵ and set the parameter $\epsilon = 1$.
DeepSHAP: DeepSHAP adapted DeepLIFT (Shrikumar et al., 2016) to approximate pixel-wise Shapley values for the input image (Lundberg & Lee, 2017). We used the code released by (Lundberg & Lee, 2017).
LIME: LIME (Ribeiro et al., 2016) trained an interpretable model to compute the attribution for each super-pixel. We used the code released by (Ribeiro et al., 2016).
Grad-CAM: Grad-CAM (Selvaraju et al., 2017) was similar to CAM (Zhou et al., 2016). Grad-CAM used gradients over the feature map, instead of the parameters of the fully connected layer. Since Grad-CAM computed attribution maps for intermediate-layer features, we used the Shapley value of each unit in the intermediate-layer feature to evaluate Grad-CAM.

J Analysis of evaluation results of our metric and previous studies

In the paragraph “Seemingly contradictory with previous metrics, but actually not.” of the paper, we have discussed the seeming conflict between the evaluation results of our metric and ROAR (Hooker et al., 2019). In this section, we provide more discussions and visualization results about this issue. ROAR removes pixels with the highest attribution values, and retrains the DNN using images after removal. In order to analyze the effect of the removal, we trained the AlexNet using the Pascal VOC 2012 dataset, and we used GB, LRP, Grad-CAM, LIME to generate attribution maps. Then we remove 25%, 50%, 75% pixels with the highest attribution values from the image. We show images during the pixel-removing process.

As Figure 11 shows, some explanation methods, such as GB and LRP, usually assign high attributions on edge pixels. Therefore, when we remove pixels with high attributions, we firstly remove edge pixels. In this case, the removal does not affect the semantic features significantly. In comparison, some methods, such as Grad-CAM and LIME, usually assign high attributions on pixels in smooth regions. In this case, removing pixels usually introduces unrealistic features (*e.g.* new dot patterns caused by the masked pixels), which usually affects the retraining of the DNN. Therefore, ROAR has a partiality towards explanation methods similar to Grad-CAM. In comparison, our metric evaluates explanation results based on the Shapley value, which satisfies four desirable properties for explanations, which ensures that our metric can measure the objectiveness of explanation results in a more convincing manner than ROAR.

Besides, we found that GI sometimes outperformed IG. It was because in image datasets, zero may not be a good baseline value for integral. Moreover, IG only considers a single integral path from the baseline input to the original input, which also affects the attribution map generated by the IG. In comparison, as discussed in (Sundararajan et al., 2017), the Shapley value considers all possible integral paths. Thus, the result of IG can be significantly different from that of the Shapley value.

K Detailed results of the bias of the attribution map at the pixel level

This section provides more results of the Figure 7 in the paper, including results on ResNet-44/50/56 and VGG-19, in Figure 12. Furthermore, the following tables provided explicit numbers of the bias of the attribution map at the pixel level.

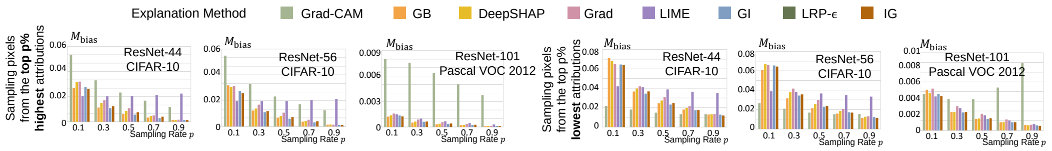

Figure 12: Bias of the attribution map at the pixel level. LRP-ϵ provided attribution maps with the least bias on LeNet, AlexNet, and VGG-16/19. GI and GB outperformed other explanation methods on ResNets. We will move these results into the additional page of the main paper if the paper is accepted.

Table 3: Pixel-level bias of the attribution map of the LeNet network learned on the CIFAR-10 dataset.

Method	Grad-CAM	Grad	GI	GB	DeepSHAP	LIME	LRP	IG
top-10%	0.17256	0.05302	0.04060	0.04701	0.04924	**0.02777**	0.04164	0.04029
top-30%	0.04921	0.03090	0.01892	0.02109	0.02408	0.02753	**0.01798**	0.02108
top-50%	0.01961	0.02057	0.01126	0.01225	0.01443	0.02780	**0.01041**	0.01349
top-70%	0.01531	0.01137	0.00671	0.00746	0.00902	0.02888	**0.00634**	0.00801
top-90%	0.01785	0.00314	**0.00240**	0.00300	0.00334	0.02996	0.00277	0.00271
bottom-10%	0.05317	0.06326	0.04907	0.06730	0.05755	**0.03522**	0.04271	0.04588
bottom-30%	0.04510	0.03948	**0.02626**	0.03224	0.03100	0.03522	0.02198	0.02666
bottom-50%	0.04732	0.02840	0.01767	0.02126	0.02047	0.03411	**0.01481**	0.01831
bottom-70%	0.04693	0.01886	0.01270	0.01551	0.01462	0.03270	**0.01086**	0.01249
bottom-90%	0.04022	0.01026	0.00808	0.01025	0.00882	0.03109	**0.00708**	0.0716

Table 4: Pixel-level bias of the attribution map of the ResNet-20 network learned on the CIFAR-10 dataset.

Method	Grad-CAM	Grad	GI	GB	DeepSHAP	LIME	IG
top-10%	0.07805	0.04692	0.03962	0.04801	0.04702	**0.02771**	0.03814
top-30%	0.04241	0.02552	**0.01522**	0.01848	0.02175	0.02769	0.01773
top-50%	0.02678	0.01616	**0.00795**	0.00943	0.01241	0.02885	0.01075
top-70%	0.01770	0.00762	**0.00416**	0.00425	0.00643	0.03071	0.00577
top-90%	0.01207	0.00288	0.00283	0.00336	0.00270	0.03264	**0.00254**
bottom-10%	**0.02858**	0.06610	0.06449	0.06726	0.06866	0.04067	0.06344
bottom-30%	**0.02363**	0.04272	0.03437	0.03833	0.03962	0.03987	0.03643
bottom-50%	**0.01930**	0.03117	0.02296	0.02704	0.02722	0.03835	0.02458
bottom-70%	**0.01556**	0.02168	0.01726	0.02053	0.01992	0.03605	0.01750
bottom-90%	0.01219	0.01357	0.01275	0.01513	0.01346	0.03409	**0.01193**

Table 5: Pixel-level bias of the attribution map of the ResNet-32 network learned on the CIFAR-10 dataset.

Method	Grad-CAM	Grad	GI	GB	DeepSHAP	LIME	IG
top-10%	0.08386	0.04836	0.04126	0.03984	0.04773	**0.02900**	0.03972
top-30%	0.05046	0.02600	**0.01641**	0.01728	0.02247	0.02848	0.01848
top-50%	0.03311	0.01639	**0.00874**	0.00972	0.01289	0.02950	0.01124
top-70%	0.02176	0.00793	**0.00459**	0.00565	0.00683	0.03122	0.00617
top-90%	0.01360	0.00268	0.00264	0.00242	0.00260	0.03299	**0.00233**
bottom-10%	**0.02653**	0.06535	0.06389	0.07271	0.06787	0.04138	0.06310
bottom-30%	**0.02260**	0.04217	0.03433	0.03612	0.03917	0.04044	0.03597
bottom-50%	**0.01825**	0.03064	0.02304	0.02359	0.02686	0.03875	0.02421
bottom-70%	**0.01410**	0.02135	0.01726	0.01733	0.01963	0.03651	0.01720
bottom-90%	**0.01064**	0.01331	0.01254	0.01214	0.01307	0.03447	0.01163

Table 6: Pixel-level bias of the attribution map of the ResNet-44 network learned on the CIFAR-10 dataset.

Method	Grad-CAM	Grad	GI	GB	DeepSHAP	LIME	IG
top-10%	0.08004	0.04795	0.04155	0.04055	0.04751	**0.03048**	0.03940
top-30%	0.04945	0.02576	**0.01626**	0.01698	0.02254	0.03035	0.01848
top-50%	0.03489	0.01609	**0.00848**	0.00935	0.01307	0.03087	0.01127
top-70%	0.02517	0.00748	**0.00433**	0.00502	0.00686	0.03189	0.00608
top-90%	0.01774	0.00248	0.00228	0.00230	0.00226	0.03357	**0.00206**
bottom-10%	**0.02186**	0.06510	0.06452	0.07143	0.06802	0.04198	0.06398
bottom-30%	**0.01812**	0.04196	0.03449	0.03663	0.03977	0.04098	0.03673
bottom-50%	**0.01488**	0.03082	0.02323	0.02448	0.02736	0.03877	0.02481
bottom-70%	**0.01325**	0.02159	0.01752	0.01811	0.01989	0.03643	0.01761
bottom-90%	0.01343	0.01353	0.01282	0.01293	0.01323	0.03487	**0.01191**

Table 7: Pixel-level bias of the attribution map of the ResNet-56 network learned on the CIFAR-10 dataset.

Method	Grad-CAM	Grad	GI	GB	DeepSHAP	LIME	IG
top-10%	0.08231	0.04716	0.04129	0.04780	0.04601	**0.02956**	0.03902
top-30%	0.04937	0.02537	**0.01659**	0.01855	0.02089	0.02932	0.01799
top-50%	0.03486	0.01607	**0.00900**	0.01022	0.01215	0.02970	0.01093
top-70%	0.02584	0.00791	**0.00482**	0.00569	0.00652	0.03089	0.00611
top-90%	0.01880	0.00212	0.00194	0.00214	0.00234	0.03220	**0.00185**
bottom-10%	**0.02661**	0.06625	0.06551	0.06065	0.06727	0.03882	0.06436
bottom-30%	**0.02093**	0.04164	0.03449	0.03158	0.03779	0.03828	0.03567
bottom-50%	**0.01689**	0.03001	0.02299	0.02129	0.02574	0.03661	0.02380
bottom-70%	**0.01524**	0.02086	0.01712	0.01587	0.01868	0.03495	0.01692
bottom-90%	0.01549	0.01300	0.01237	**0.01146**	0.01267	0.03338	0.01150

Table 8: Pixel-level bias of the attribution map of the ResNet-50 network learned on the Pascal VOC 2012 dataset.

Method	Grad-CAM	Grad	GI	GB	DeepSHAP	LIME	IG
top-10%	0.06494	0.00766	0.00728	0.00633	0.00738	**0.00613**	0.00701
top-30%	0.04189	0.00437	0.00345	**0.00276**	0.00359	0.00374	0.00369
top-50%	0.02715	0.00297	0.00213	**0.00170**	0.00223	0.00223	0.00241
top-70%	0.01683	0.00173	0.00138	**0.00113**	0.00144	0.00125	0.00152
top-90%	0.01022	0.00060	0.00060	**0.00053**	0.00062	0.00104	0.00060
bottom-10%	0.02107	0.00831	**0.00824**	0.00850	0.00839	0.00853	0.00811
bottom-30%	0.01964	0.00500	0.00417	**0.00386**	0.00432	0.00591	0.00444
bottom-50%	0.01674	0.00354	0.00269	**0.00244**	0.00280	0.00432	0.00295
bottom-70%	0.01284	0.00228	0.00189	**0.00171**	0.00194	0.00311	0.00196
bottom-90%	0.00917	0.00117	0.00112	0.00111	0.00114	0.00191	**0.00108**

Table 9: Pixel-level bias of the attribution map of the ResNet-101 network learned on the Pascal VOC 2012 dataset.

Method	Grad-CAM	Grad	GI	GB	DeepSHAP	LIME	IG
top-10%	0.03996	0.00790	0.00678	**0.00610**	0.00685	0.00744	0.00622
top-30%	0.03791	0.00438	0.00307	**0.00260**	0.00318	0.00495	0.00320
top-50%	0.03184	0.00291	0.00185	**0.00158**	0.00194	0.00339	0.00206
top-70%	0.02514	0.00175	0.00121	**0.00104**	0.00127	0.00228	0.00131
top-90%	0.01884	0.00067	0.00055	**0.00049**	0.00056	0.00147	0.00054
bottom-10%	0.00753	0.00861	0.00754	0.00839	0.00767	**0.00700**	0.00710
bottom-30%	0.00649	0.00496	**0.00369**	0.00377	0.00381	0.00460	0.00381
bottom-50%	0.00643	0.00343	**0.00237**	0.00238	0.00245	0.00311	0.00251
bottom-70%	0.00883	0.00225	**0.00169**	0.00169	0.00173	0.00206	**0.00169**
bottom-90%	0.01384	0.00117	0.00104	0.00112	0.00105	0.00132	**0.00094**

Table 10: Pixel-level bias of the attribution map of the AlexNet network learned on the Pascal VOC 2012 dataset.

Method	Grad-CAM	Grad	GI	GB	DeepSHAP	LIME	LRP	IG
top-10%	0.05403	0.00744	0.00505	0.00646	0.00845	0.00712	**0.00453**	0.00473
top-30%	0.03467	0.00434	0.00243	0.00296	0.00447	0.00501	**0.00197**	0.00256
top-50%	0.02459	0.00298	0.00149	0.00182	0.00295	0.00354	**0.00119**	0.00168
top-70%	0.01711	0.00176	0.00094	0.00120	0.00209	0.00232	**0.00080**	0.00104
top-90%	0.01115	0.00060	**0.00035**	0.00058	0.00140	0.00143	0.00043	0.00037
bottom-10%	0.01026	0.00906	0.00638	0.00876	0.00438	0.00703	**0.00430**	0.00614
bottom-30%	0.00978	0.00534	0.00327	0.00400	0.00215	0.00480	**0.00202**	0.00336
bottom-50%	0.00803	0.00372	0.00215	0.00253	0.00131	0.00332	**0.00129**	0.00224
bottom-70%	0.00610	0.00239	0.00151	0.00178	**0.00075**	0.00221	0.00091	0.00150
bottom-90%	0.00690	0.00124	0.00093	0.00111	**0.00030**	0.00141	0.00056	0.00084

Table 11: Pixel-level bias of the attribution map of the VGG-16 network learned on the Pascal VOC 2012 dataset.

Method	Grad-CAM	Grad	GI	GB	DeepSHAP	LIME	LRP	IG
top-10%	0.02984	0.00764	0.00656	0.00563	0.00768	0.00540	**0.00461**	0.00625
top-30%	0.01876	0.00405	0.00292	0.00253	0.00310	0.00317	**0.00159**	0.00304
top-50%	0.01365	0.00267	0.00176	0.00154	0.00187	0.00205	**0.00092**	0.00195
top-70%	0.00947	0.00162	0.00115	0.00098	0.00132	0.00153	**0.00065**	0.00124
top-90%	0.00650	0.00061	0.00051	**0.00038**	0.00089	0.00163	0.00044	0.00051
bottom-10%	0.00767	0.00869	0.00768	0.00865	0.00396	0.00805	**0.00288**	0.00742
bottom-30%	0.00708	0.00483	0.00372	0.00406	0.00173	0.00588	**0.00117**	0.00384
bottom-50%	0.00654	0.00330	0.00238	0.00261	0.00106	0.00446	**0.00072**	0.00251
bottom-70%	0.00540	0.00219	0.00170	0.00185	0.00075	0.00327	**0.00054**	0.00171
bottom-90%	0.00509	0.00120	0.00107	0.00122	0.00041	0.00241	**0.00040**	0.00101

Table 12: Pixel-level bias of the attribution map of the VGG-19 network learned on the Pascal VOC 2012 dataset.

Method	Grad-CAM	Grad	GI	GB	DeepSHAP	LIME	LRP	IG
top-10%	0.02564	0.00761	0.00672	0.00580	0.00802	0.00550	**0.00468**	0.00633
top-30%	0.01991	0.00403	0.00301	0.00261	0.00337	0.00308	**0.00157**	0.00311
top-50%	0.01517	0.00263	0.00181	0.00159	0.00205	0.00171	**0.00091**	0.00197
top-70%	0.01139	0.00161	0.00118	0.00102	0.00143	0.00097	**0.00064**	0.00126
top-90%	0.00799	0.00062	0.00053	**0.00043**	0.00096	0.00124	0.00045	0.00052
bottom-10%	0.00932	0.00867	0.00782	0.00871	0.00442	0.00890	**0.00236**	0.00743
bottom-30%	0.00764	0.00480	0.00378	0.00406	0.00193	0.00621	**0.00095**	0.00385
bottom-50%	0.00683	0.00326	0.00243	0.00260	0.00120	0.00472	**0.00059**	0.00252
bottom-70%	0.00630	0.00217	0.00173	0.00184	0.00084	0.00354	**0.00046**	0.00173
bottom-90%	0.00649	0.00119	0.00109	0.00120	0.00043	0.00245	**0.00036**	0.00101

L Comparison between our metric and the evaluation based on DeepSHAP

In Figure 2 in the paper, we have shown that the attribution map of DeepSHAP can be significantly different from the relatively accurate Shapley value. In this section, we will provide an experiment to further show the unreliability of DeepSHAP.

However, since the proposed metric just represents the bias of the entire attribution distribution over all pixels, instead of measuring the attribution value for a specific pixel, it is not appropriate to directly compare our metric with DeepSHAP. Therefore, we design another metric, which measured the average attribution over the top p-% (10%, 30%, 50%, 70%, 90%) attribution estimated by DeepSHAP. In this way, such a metric was comparable with our metric.

In the experiment, we evaluate the accurate Shapley value, which was computed using NP-hard computation, by using both our metric and the metric based on DeepSHAP. Since the Shapley value itself is an unbiased attribution method, if our metric showed a lower bias than the metric based on DeepSHAP, then we considered our metric was more convincing.

For implementation, we constructed an MLP-5, and trained it using the TV news channel commercial detection dataset (Vyas et al., 2014). Each sample in this dataset only contains 10 input variables, which made it possible to accurately compute the Shapley value for each input variable.

The evaluation result is given in Table 13. We found that the bias of our metric was significantly lower than the bias of the metric based on DeepSHAP. Thus, our metric is more reliable than the metric based on DeepSHAP.

Table 13: Bias of our metric and the metric based on DeepSHAP when evaluating the accurate Shapley value.

Metric	Sample top-10%	Sample top-30%	Sample top-50%	Sample top-70%	Sample top-90%
Our metric	**0.0102**	**0.0030**	**0.0024**	**0.0018**	**0.0015**
Metric based on DeepSHAP	0.1944	0.0990	0.0774	0.0550	0.0428

M Verification of assumptions in the first and the third paragraphs of page 5.

In Assumption 1 and Lemma 1, we made two assumptions to design the evaluation metric. In this section, we will conduct experiments to verify these two assumptions. For the convenience of readers, we will rewrite the two assumptions here. In the first paragraph of page 5, we assume that the attribution value of each pixel follows a Gaussian distribution, and different pixels share the same variance. In the third paragraph of page 5, we assume that the approximated Shapley value of each pixel follows a Gaussian distribution.

For the assumption in the first paragraph of page 5, we selected attribution values from a region with 5×5 pixels in 100 images. Then, we computed the average value and the variance using these $5 \times 5 \times 100 = 2500$ attribution values to test whether they follow the Gaussian distribution. Furthermore, in this experiment, we randomly selected three different regions in these images. We compared the variance computing using pixels from different regions, to check whether these regions share a same variance. Specifically, we conducted this experiment using the ResNet-20 trained using the CIFAR-10 dataset, and we used the Gradient×Input to generate attribution values. Table 14 shows the result. We found that **distributions on these three different regions had similar mean values and similar variances.** *I.e.,* the mean value of attribution values from different regions all approximately equaled to 0, and variance values from different regions all approximately equaled to 0.7 in most cases. Therefore, the assumption in Line 181 was verified.

Table 14: Mean and variance of attribution values generated by Gradient×Input

Region	Region 1	Region 2	Region 3
Mean	-0.0129	0.0086	0.0004
Variance	0.7243	0.6894	0.7719

For the assumption in the third paragraph of page 5, we followed the above experimental setting, and used the approximated Shapley value as the attribution value. Table 15 shows that the approximated Shapley values from different regions also share similar mean values and similar variances, which approximately equaled to 0.02 in most cases. Therefore, the assumption was verified.

Table 15: Mean and variance of the approximated Shapley value

Region	Region 1	Region 2	Region 3
Mean	0.0094	0.0092	0.0107
Variance	0.0019	0.0017	0.0018

N Untrustworthiness of the ground-truth explanation constructed in (Yilun et al., 2022; Jasmijn et al., 2021)

In the first three paragraphs of Section 3, we have conducted an experiment to show that the ground-truth explanation created by (Yang & Kim, 2019; Camburu et al., 2019) were not trustworthy. Actually, several other studies (Yilun et al., 2022; Jasmijn et al., 2021) also proposed similar evaluation methods as (Yang & Kim, 2019; Camburu et al., 2019). In this section, we will conduct a **new experiment** to show that the ground-truth attributions constructed in (Yilun et al., 2022; Jasmijn et al., 2021) are also untrustworthy. We created a binary-classification task. The DNN was trained to classify whether there was a specific pattern in the image. Here, we used the Pascal VOC 2012 dataset in this experiment, used the logo of the Pascal VOC 2012 dataset as the pattern. Specifically, each image in this dataset was randomly assigned as the positive sample with the probability of 0.5, and we randomly replaced a region with 56×56 pixels with the pattern for each positive sample. Ideally, the DNN was supposed to only use pixels in the pattern for the classification. We trained an AlexNet using the constructed dataset, and estimated Shapley value using images with the pattern. We computed the ratio of Shapley values in the pattern to Shapley values in the whole image, as $\frac{\sum_{i\in\text{pattern}}|A_i^{\text{shap}}|}{\sum_{i\in\text{image}}|A_i^{\text{shap}}|}$. We found that the ratio was not equal to 1, and the average ratio over all images was **0.898**. Therefore, the model did not only use pixels in the pattern for the inference. Besides, we also found that Shapley values of different pixels in the pattern were different with each other, so we should not consider all pixels in the pattern were used for the inference. This indicated the untrustworthiness of such evaluation methods. Therefore, the constructed ground-truth in these studies was unreliable.