Augmentative eXplanation and the Distributional Gap of Confidence Optimization Score

Erico Tjoa 1 2  Hong Jing Khok 1  Tushar Chouhan 1  Guan Cuntai 1

Abstract

This paper introduces the Confidence Optimization (CO) score to directly measure the contribution of heatmaps/saliency maps to the classification performance of a model. Common heatmap generation methods used in the eXplainable Artificial Intelligence (XAI) community are tested through a process we call the Augmentative eXplanation (AX). We find a surprising gap in CO scores distribution on these heatmap methods. The gap potentially serves as a novel indicator for the correctness of deep neural network (DNN) prediction. We further introduce Generative AX (GAX) method to generate saliency maps capable of attaining high CO scores. Using GAX, we also qualitatively demonstrate the unintuitiveness of DNN architectures.

1. Introduction

Recent issues on the usage of responsible AI have been highlighted by large companies like Google (Lakshmanan, 2021) and Meta (Pesenti, 2021); this may reflect the increasing demand for eXplainable Artificial Intelligence (XAI). Artificial intelligence (AI) and machine learning (ML) models have been developed with various levels of transparency and interpretability, as shown in a number of surveys that provide different taxonomies and classifications (Adadi & Berrada, 2018; Došilović et al., 2018; Gilpin et al., 2018; Tjoa & Guan, 2020a). Popular XAI methods include post-hoc methods such as Local Interpretable Model-agnostic Explanations (LIME) (Ribeiro et al., 2016) and SHapley Additive exPlanations (SHAP) that uses a game-theoretical concept (Lundberg & Lee, 2017). Many heatmap-generating XAI methods have also been developed for DNN, in particular Class Activation Mappings (CAM) (Zhou et al., 2016; Selvaraju et al., 2016), Layerwise Relevance Propagation (LRP) (Bach et al., 2015) and many other well-known methods, as listed in aforementioned surveys papers. These methods are appealing because heatmap-like attributions are easy to understand. Other methods directly attempt to visualize the inner working of a DNN (Zeiler & Fergus, 2014; Olah et al., 2017; 2020). They visualize the responses of an individual neuron or a collection of neurons when a DNN is fed with data samples. There are certainly other remarkable ways to investigate interpretability and explainability that are not covered here.

Note: we will use heatmaps and saliency maps interchangeably in this paper.

Several efforts have been dedicated to quantitatively measure the quality of heatmaps and other explanations. For example, heatmaps have been measured by their potentials to improve object localization performance (Zhou et al., 2016; Selvaraju et al., 2016). The pointing game (Fong & Vedaldi, 2017; Rebuffi et al., 2020) is another example where localization concept is used to quantify XAI’s performance. Others can be found e.g. in (Tjoa & Guan, 2020b); here, we pay particular attention to the suggestion presented in the paper that introduced the “most relevant first” (MORF) framework to quantify the explainability of heatmaps (Samek et al., 2017). It suggests that there is a difference between computational relevance and human relevance suggesting that objects which algorithms find salient may not be necessarily salient for a human observer.

However, heatmaps have been said to not “[tell] us anything except where the network is looking at” (Rudin, 2019). To make heatmaps more useful, we perform the Augmentative eXplanation (AX) process: combine each image \( x \) with its heatmap \( h \), and hope that predictive probability is improved, e.g. if \( f(x) \) gives a 60\% probability of making a correct prediction, we find \( h \) such that \( f(x + h) \) yields 65\%. We show such improvement does not happen in general to existing XAI methods, since heatmaps are usually not designed to explicitly improve prediction computationally. We will thus introduce the Generative AX (GAX) that is explicitly designed to improve prediction. Confidence Optimization (CO) score is introduced to attain such improvement and simultaneously used as the measure of computational relevance. Briefly speaking, CO...
score is a weighted difference between raw output values before and after heatmaps/attributions modify the images \( x + h \). The metric assigns a positive/negative score if \( x + h \) increases/decreases the probability of making the correct prediction. DISCLAIMER: we explicitly avoid making references to human relevance, unlike existing heatmap methods which often demonstrate localization capability etc that apparently match human intuition.

This paper is arranged as the following. In the next section, AX and Generative AX (GAX) is demonstrated through a two-dimensional toy example. Explicit form of heatmaps/attribute values can be obtained in the toy example, useful for lower level analysis and direct observation. The following section describes dataset pre-processing, computation of CO scores for existing XAI methods, formal definition GAX process and the results. We then present our results, starting with the novel finding: distribution gap as correctness indicators, CO scores distribution for common XAI methods, followed by high scores attained GAX heatmaps and finally qualitative aspects of the methods. All codes are available in https://github.com/ericotjo001/explainable_ai/tree/master/gax.

2. Formulation in Low Dimensional Space

The application presented in this paper is based on the following concept. We illustrate the idea using binary classification of data sample \( x \in \mathbb{R}^2 \), a 2D toy example. Let \( y = W^{-1}x \) where \( y \in \mathbb{R}^2 \) and \( W \in \mathbb{R}^{2\times 2} \) is invertible. Let the true label/category of sample \( x \) be \( c = \arg\max_i y_i \) so that it is compatible with one-hot encoding usually used in a DNN classification task. Conventions:

1. \textit{Output space} \( Y \). Let the output variable be \( y = a_1(1) + a_2(0) \), clearly an element of a vector space. The shape of this vector is the same as the output shape of the last fully connected layer for the standard binary classification. Class prediction can be performed in the winner-takes-all manner, for example, if \( a_1 = 1, a_2 = 0 \), then the label is \( c = \arg\max_i y_i = 1 \). If \( a_1 = 0.1, a_2 = 0.5 \), then \( c = 2 \). \textit{Basis of} \( Y \) is \( B_Y = \{ y^{(1)} = (1), y^{(2)} = (0) \} \).

2. \textit{Sample space} \( X \) is a vector space with the corresponding basis \( B_X = \{ Wy : y \in B_Y \} = \{ x^{(1)} = Wy^{(1)}, x^{(2)} = Wy^{(2)} \} \) so \( x = a_1 x^{(1)} + a_2 x^{(2)} \in X \).

3. \textit{Pixelwise sample space} is the same sample space, but we specifically distinguish it as the sample space with the canonical basis. We will need this later, because pixelwise space has “human relevance”, since human observers perceive the components (pixels) directly, rather than automatically knowing the underlying structure (i.e. we cannot see \( a_1, a_2 \) directly). We denote a sample in this basis with \( x = x_1(\frac{1}{2}) + x_2(\frac{1}{2}) \).

4. A heatmap or attribute vector \( h \) in this paper has the same shape as \( x \) and can be operated directly with \( x \) via component-wise addition. Thus, they also belong to sample space or the pixelwise sample space. Writing a heatmap in the sample space \( h = Ax^{(1)} + Bx^{(2)} \) is useful for obtaining a closed form expression later.

The perfect classifier, \( f \). Define \( f(x, \Theta) = \sigma(\Theta x) \) as a trainable classifier with parameters \( \Theta \in \mathbb{R}^{2\times 2} \). Let \( \Theta = W^{-1} \) and the activation \( \sigma \) be any strictly monotonic function, like the sigmoid function. Then, the classifier \( f(x) = \sigma(W^{-1}x) \in \mathbb{R}^2 \) is perfect, in the sense that, if \( a_1 > a_2 \), then \( c = \arg\max_i f_i(x) = 1 \); likewise if \( a_1 < a_2 \), then \( c = 2 \) and, for \( a_1 = a_2 \) either decision is equally probable. This is easily seen as the following: \( f(x) = \sigma(W^{-1}(a_1 x^{(1)} + a_2 x^{(2)})) = \sigma(a_1 (\frac{1}{2}) + a_2 (\frac{0}{1})) = (\sigma(a_1), \sigma(a_2)) \).

Confidence optimization score (CO score), \( s_{co} \). In this section, we show a simple explicit form of CO score for better illustration; in the experimental method section, formal definition will be given. The score increases if \( x + h \) leads to an improvement in the probability of correctly predicting label \( c \), hence the score’s definition depends on the groundtruth label. Throughout this section, for illustration, we use \( x = a_1 x^{(1)} + a_2 x^{(2)} \) with groundtruth label \( c = 1 \), i.e. \( a_1 > a_2 \). Define the CO score as

\[
s_{co}(x, h) = (\frac{1}{2}) \cdot [f(x + h) - f(x)] \tag{1}
\]

For the perfect classifier, see that \( f_1(x + h) > f_1(x) \) and \( f_2(x + h) < f_2(x) \) contribute to a larger \( s_{co} \). In other words, increasing the probability of predicting the correct label \( c = 1 \) increases the score. For \( c = 2 \), replace \( (\frac{1}{2}) \) with \( (\frac{1}{2}) \).

Augmentative explanation. AX is defined here as any modification on \( x \) by \( h \) that is intended to yield positive the CO score, i.e to increase the probability of making a correct classification. This paper mainly considers the simplest implementation, namely \( x + h \). Let us consider a few possibilities. Suppose \( \sigma = \text{LeakyReLU} \) and \( h = x \). We get \( s_{co} = (\frac{1}{2}) \cdot \left( \frac{\sigma(a_1) - \sigma(a_2)}{\sigma(a_2) - \sigma(a_1)} \right) = a_1 - a_2 > 0 \). In other words, choosing the image as the heatmap itself improves the score. However, as a heatmap or attribute vector, \( h \) is useless, since it does not provide us with any information about the relative importance of the components of \( x \) in canonical basis, which is the part of data directly visible to the observer. Even so, \( h = x \) has computational relevance to the model, since \( a_1, a_2 \) are modified in the correct direction. Our aim is to find computationally relevant \( h \) that does not score zero in “human relevance”, figuratively speaking. We therefore rule out obviously uninformative heatmap in the upcoming sections. Further, consider similar situa-
tion but set $\sigma$ to sigmoid function. Simply setting $h = x$ will no longer increase the score significantly all the time. Since sigmoid is *asymptotic*, when $a_1, a_2$ are sufficiently far away from zero, the increase will be so negligible, the heatmap will be uninformative even though the magnitude of $|a_1 - a_2|$ may be large. Hence, we use the raw DNN output in our main experiment, without sigmoid, softmax etc.

**Heatmap obtained through optimization using gradient ascent.** Recall that gradient ascent is done by $\Delta \rightarrow \Delta + dw \cdot \nabla_w \Delta$ with the choice $dw = \eta \nabla_w \Delta$, hence $\Delta + \eta \| \nabla_w \Delta \|^2 \geq \Delta$. Hence, the heatmap after $k$ steps of optimization is given by

$$h = (w + kdw) \ast x$$

(4)

To visualize the heatmap, here we use the example where $W$ is the rotation matrix $W = \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix}$. Examples of heatmaps plotted along with the input $x$ are shown in fig. 1, to be discussed in the next subsection. If $\theta = 0$, $x$ is identical to $y$, so binary classification is straightforward and requires no explanation. Otherwise, consider $\theta$ being a small deviation from 0. Such slightly rotated system is a good toy-example for the demonstration of component-wise “importance attribution”. This is because if $x$ belongs to category $c = 1$ with high $a_1$ component, then it still has a more significant first component $x_1$ after the small rotation. Thus, a heatmap that correspondingly gives a higher score to the first component is “correct” in the sense that it matches the intuition of attribute importance: high $h_1$ emphasizes the fact that high $x_1$ literally causes high $y_1$. Furthermore, if the system rotates by $\pi/4$, we see that the classification becomes harder. This is because the components $x_1$ and $x_2$ start to look more similar because $\cos \frac{\pi}{4} = \sin \frac{\pi}{4}$; and consequently, the attribution values will be less prominent as well.

2.1. Interpretability

**DISCLAIMER:** for the benefit of readers who are used to regard heatmaps as the explanation, we must emphasize that this paper does not appeal to that ideal. To reiterate, the main interpretation of our paper is to computationally optimize predictive probability using heatmaps.

**Homogenous and Heterogenous transformations.** For the lack of better words, we refer to transformations like $\theta \approx \pi/4$ or more generally $(2k + 1)\frac{\pi}{4}$ for $k = ..., -1, 0, 1, ...$ as homogenous transformations, since the components become more indistinguishable (recall: $\cos \frac{\pi}{4} = \sin \frac{\pi}{4}$). Otherwise, the transformation is called heterogenous. These definitions are given here with the intention of drawing parallels between (1) the toy data that have been homogenously transformed (hence hard to distinguish) and (2) samples in real datasets that look similar to each other, but are categorized differently due to a small, not obvious difference.

**Interpretation of attribute values for distinct non-negative components.** In the pixelwise sample space, we will be more interested in non-negative data sample $x_1, x_2 \geq 0$ since we only pass $[0, 1]$-normalized images for GAX. Fig. 1 left shows a data sample with distinct components, indicated
Augmentative eXplanation and the Distributional Gap of Confidence Optimization Score

by high $a_1 = 0.95$ component and low $a_2$. Non-negative data samples are found around $\theta \in [0, \pi/2]$. High $x_1$ value is given high $h_1$ attribution score while low $x_2$ is given a suppressed value of $h_2$ near $\theta = 0$, matching our intuition as desired. As rotation proceeds to $\pi/4$, there is a convergence between $x_1$ and $x_2$, making the components more indistinguishable. At $\theta = \pi/4$ exactly, we still see high $h_1$ that picks up high signal due to high $a_1$, also as desired. Between $\pi/4$ and $\pi/2$, rotation starts to flip the components; in fact, at $\pi/2$, $x = [0, 1]$ is categorized as $c = 1$ and $x = [1, 0]$ as $c = 2$. The attribution value $h_2$ becomes more prominent, highlighting $x_2$, also as desired for our prediction of class $c = 1$. In fig. 1 middle, decreased/increased $a_1, a_2$ are assigned less prominent $h_1, h_2$ respectively than fig. 1 left, since the model becomes less confident in its prediction, also consistent with our intuition.

The other extreme. Fig. 1 right shows $a_1$ and $a_2$ that do not differ significantly. At homogeneous transformation $\theta = \pm \frac{\pi}{4}$, heatmaps are almost equal to the input $x$. As expected, it will be difficult to pick up signals that are very similar, although very close inspection might reveal small differences that could probably yield some information (not in the scope of this paper). Other interpretations can be found in appendix More interpretations in low dimensional example.

3. Experimental Method and Datasets

In the previous section, we described how heatmap $h$ can be used to improve classification probability. More precisely, $x + h$ yields higher confidence in making a correct prediction compared to $x$ alone when used as the input to the model $f$. We apply the same method to real dataset ImageNet (Deng et al., 2009) and Chest X-Ray Images (Pneumonia) from Kaggle (Mooney, 2018). The Pneumonia dataset needs reshuffling, since Kaggle’s validation dataset consists of only 16 images for healthy and pneumonia cases combined. We combined the training and validation datasets and then randomly draw 266/1083 healthy and 790/3093 pneumonia images for validation/ training. There are 234/390 healthy/pneumonia images in the test dataset. Images are all resized to $256 \times 256$. The X-Ray images are black and white, so we stack them to 3 channels. Images from ImageNet are normalized according to suggestion in the pytorch website, with mean $= [0.485, 0.456, 0.406]$, std $= [0.229, 0.224, 0.225]$.

For both datasets, we use pre-trained models Resnet34 (He et al., 2016) and AlexNet (Krizhevsky, 2014) available in Pytorch. The models are used on ImageNet without fine-tuning. Resnet34 is fine-tuned for Pneumonia binary classifications, where the first 8 modules of the pre-trained model (according to pytorch’s arrangement) are used, plus a new fully-connected (FC) layer with two output channels at the end. Similarly, for Alexnet, the first 6 modules are used with a two-channel FC at the end. For Resnet34, we will use Resnet34_1 and Resnet34_sub respectively trained to achieve 99% and 80% validation accuracies for comparison. The same targets were specified for Alexnet, but only 80% validation accuracy was achieved, thus only Alexnet_sub will be used. Adam optimizer is used with learning rate $0.001, \beta = (0.5, 0.999)$. The usual weight regularization is not used during optimization i.e. in pytorch’s Adam optimizer, weight decay is set to zero because we allow zero attribution values in large patches of the images. No number of epochs are specified. Instead, training is stopped after the max number of iterations (240000) or the specified validation accuracy is achieved after 2400 iterations have passed. At each iteration, samples are drawn uniform-randomly with batch size 32.

CO scores on existing XAI methods via AX process. Denote the deep neural network as DNN, define the CO score as the weighted difference between the predictive scores altered by AX process and the original predictive scores,

$$s_{co}(x, h) = \kappa \cdot [DNN(x + h) - DNN(x)] \quad (5)$$

where $\kappa \in \mathbb{R}^C$ is defined as the score constants, $C$ the number of classes, $\kappa_j = 1$ if the groundtruth belongs to label/category $j$ and $\kappa_i = -1/(C - 1)$ for all $i \neq j$. This equation is the general form of eq. (1). In our implementation, each DNN’s output is raw, i.e. last layer is FC with $C$ channels without softmax layer etc. A heatmap $h$ that yields $s_{co} = 0$ is uninformative (see appendix). We compute CO scores for heatmaps generated by six different existing heatmap-based XAI methods (all available in PyTorch Captum), namely, Saliency (Simonyan et al., 2014), Input*Gradient (Shrikumar et al., 2016), Layer GradCAM (Selvaraju et al., 2016), Decon- volution (Zeiler & Fergus, 2014), Guided Backpropagation (Springenberg et al., 2015) and DeepLift (Shrikumar et al., 2017). Each heatmap is generated w.r.t predicted target, not groundtruth e.g. if $y_{pred}=$DNN$(x)$ predicts class $n$, then $h=\text{DeepLIFT}(\text{net}).\text{attribute}(\text{input}, \text{target}=n)$ in Pytorch Captum notation. Then normalization is applied $h \rightarrow h/\max(|h|)$ before we perform the AX process. Note: For ImageNet, $C = 1000$, chest X-Ray, $C = 2$. We also consider $f(x + h)$, where * denotes component-wise multiplication. The idea is generally to interact $h$ with $x$ so that, for any interaction $g$, higher probability of correct prediction is achieved by $f(g(x, h))$; see appendix for their results.

Achieving high $s_{co}$ with GAX. Here, GAX is the $x + h$ AX process where heatmaps $h = \tanh(x \ast w)$ are generated by training parameters $w$ to maximize $s_{co}$. Maximizing $s_{co}$ indefinitely is impractical, and thus we have chosen $s_{co} = 48$ for ImageNet dataset, a score higher than most $s_{co}$ attained by existing XAI methods we tested in this experiment. Tanh activation is used both to ensure non-linearity and to en-
Figure 2. Distribution of CO scores obtained through AX process on existing XAI methods. All distributions show gaps between CO scores of data whose classes are correctly and wrongly predicted (e.g. red arrows). Correct prediction tends to yield higher CO scores. (A) Resnet34 on Pneumonia dataset. (B) Similar Resnet34 architecture on Pneumonia dataset, but with less fine-tuning (Resnet34_sub). (C) Resnet34 on ImageNet.

Table 1. Fine-tuning results for pre-trained models on Chest X-Ray Pneumonia test dataset.

|                  | Resnet34_1 | Resnet34_sub | Alexnet_sub |
|------------------|------------|--------------|-------------|
| accuracy         | 0.800      | 0.636        | 0.745       |
| precision        | 0.757      | 0.632        | 0.726       |
| recall           | 1.000      | 1.000        | 0.951       |
| val. acc.        | 0.990      | 0.800        | 0.800       |

Sure that the heatmap is normalized to $[-1, 1]$ range, so that we can make a fair comparison with existing heatmap-based XAI methods. For ImageNet, 10000 data samples are randomly drawn from the validation dataset for evaluating GAX. For pneumonia, all data samples are used. Optimization is done with Adam optimizer with learning rate 0.1, $\beta = (0.9, 0.999)$. This typically takes less than 50 steps of optimization, a few seconds per data sample using a small GPU like NVIDIA GeForce GTX 1050.

**Similarity loss and GAX bias.** In our implementation, we minimize $-s_{co}$. However, this is prone to producing heatmaps that are visually imperceptible from the image. Since $w$ is initialized as an array of 1s with exactly the same shape $(c, h, w) = (3, 256, 256)$ as $x$, the initial heatmap is simply $h = w \times x = x$. Possibly, small changes in $w$ over the entire pixel space is enough to cause large changes in the prediction, reminiscent of adversarial attack (Szegedy et al., 2014; Akhtar & Mian, 2018). We solve this problem by adding the similarity loss, penalizing $h = x$. The optimization is now done by minimizing the modified loss, which is negative CO score plus similarity loss

$$loss = -s_{co} + l_s \left\langle \frac{(h - x + \epsilon)^2}{x + \epsilon} \right\rangle^{-1}$$

where $l_s = 100$ is the similarity loss factor. $\langle X \rangle$ computes the average over all pixels. Division $/$ and square $^2$ are performed component/pixel-wise. Pixel-wise division by $x$ normalizes the pixel magnitude, so that small pixel values can contribute more significantly to the average value. The small term $\epsilon = 10^{-4}$ is to prevent division by zero and possibly helps optimization by ensuring that zero terms do not make the gradients vanish. Furthermore, for X-Ray images, with many zeros (black region), the similarity factor seems insufficient, resulting in heatmaps that mirror the input images. GAX bias is added for the optimization to work, so that $h = w \times x + b$, where $b$ is 0.01 array of the same shape $(c, h, w)$ as well. Note: the similarity loss is positive, since $x$ used here is $[0, 1]$ normalized (by comparison, the standard Resnet34 normalization can result in negative pixels).

4. Results and Discussions

Recall that we use pre-trained models for ImageNet. For pneumonia dataset, the predictive results of fine-tuning models are shown in table 1. AX and GAX processes will be applied on top of these models.

4.1. Gaps in CO Scores Distribution

Here, we present the main novel finding: a novel gap distribution. AX process neither specifies any formulas nor optimizes any losses to distinguish correct predictions from the wrong ones, but fig. 2 shows distinct gaps between them (shown by the red arrows). This is a surprising find since none of the set up is designed to capture the correctness of prediction. This indicates that some existing XAI methods possess information to distinguish between correct
and wrong predictions more than the others. In the absence of such information, we expect to see uniformly random distribution of scores, yet we have observed distinct distributional gaps. As such, CO score might be a new indicator to help support classification decision made by the particular architecture.

Furthermore, the extent of CO score distribution gap is clearly dependent on the dataset and DNN architecture. As it is, the discriminative capability of different XAI methods is thus only comparable within the same system of architecture and dataset. ImageNet dataset shows a smaller gap compared to pneumonia dataset and the largest gap in ImageNet is produced by the Saliency method, as seen in fig. 2(C). By comparison, the largest gap in pneumonia dataset is produced by Deconvolution. Comparing fig. 2 (A) and (B), the gaps appear to be wider when DNN is better trained. Further investigation is necessary to explain the above observations, but, to leverage this property, users are encouraged to test AX process on different XAI methods to find the particular method that shows the largest gap. Once the XAI method is determined, it can be used as a supporting tool and indicator for the correctness of prediction.

4.2. Improvement in Predictive Probability with GAX

The higher the CO scores are, the better is the improvement in predictive probability. Is it possible to achieve even higher improvement, i.e. higher CO score? We have seen that fig. 2 shows the boxplots of CO scores for AX process applied on all XAI methods we tested in this experiment; histograms applied on select XAI methods are also shown. Different XAI methods achieve different CO scores. For pneumonia dataset, very high CO scores (over 80) are attained by Deconvolution methods. For ImageNet, highset CO scores attained are around 10. To attain even higher scores, Generative AX (GAX) will be used.

Using GAX on ImageNet, $s_{co} \geq 48$ can be attained as shown in fig. 3, where the time evolution of CO score for each image is represented by a curve. For Resnet34, most of the images attains $s_{co} \geq 48$ within 50 iterations. Alexnet GAX optimization generally takes more iterations to achieve the target. High $s_{co}$ implies high confidence in making the correct prediction. We have thus obtained heatmaps and attribution values with computational relevance i.e. they can be used to improve the model’s performance. Note: (1) all images tested do attain the target $s_{co}$ (not shown), although some of the images took a few hundreds iterations (2) we exclude images where predictions are made incorrectly by the pre-trained or fine-tuned model. By comparison, in general, using heatmaps derived from existing methods for AX process does not yield positive CO scores i.e. does not improve predictive probability for the correct class (see fig. 2). Furthermore, for ImageNet, typically, $s_{co} \leq 10$. Other boxplots are shown in appendix fig. 7.

4.3. Qualitative Assessment of GAX

Heatmaps in GAX are obtained through a process optimization through a finite number of time steps. We provide matplotlib-based graphic user interface (GUI) for users to observe heatmaps features through GAX; see fig. 4(A). This provides users some information about the way the DNN architecture perceives input images. But, how exactly can user interpret this? Recall that the main premise of this paper is the computational relevance. The methods introduced in this paper are designed to generate heatmaps that improve the confidence in predicting the correct label numerically. The visual cues generated by the GAX heatmaps only show which pixels can be increased or decreased in intensity to give higher probability of making the correct prediction. As expected, we do not see any humanly interpretable heatmaps; for example, we do not observe obvious object localization, see fig. 4(B-E).

Useful patterns? Pneumonia datasets consist of chest X-Ray of healthy patients and patients with several types of pneumonia with different recognizable patterns. Bacterial pneumonia has a focal lobar consolidation, while viral pneumonia has diffuse interstitial patterns; normal chest X-Ray has neither. This turns out to affect the shape of GAX heatmaps. Fig. 4(B) shows a typical normal Chest X-Ray pattern. By comparison, fig. 4(C) shows a heatmap generated on bacterial pneumonia. In the latter, we see the drastic change in the heatmap features, especially high-intensity stripes around the lobar consolidation. There is a possibility that novel class-discriminatory patterns lie hidden within heatmaps generated by GAX.

Two causes of concerns and future works to address
Augmentative eXplanation and the Distributional Gap of Confidence Optimization Score

Figure 4. (A) GAX dynamic heatmaps displayed with a slider for users to observe the evolution of heatmaps through time steps. (B-E) GAX heatmaps generated on Resnet34 for (B) healthy chest X-Ray and (C) chest X-Ray of a patient with bacterial pneumonia; and for ImageNet images (D) a sheep dog image and (E) a planetarium image. Three heatmaps for each image correspond to the attribution values assigned to R, G and B color channels respectively. “Abs max” specifies the maximum absolute value attained by the heatmap throughout all three channels (max is 1, due to Tanh activation). Positive/negative heatmap or attribution values (red/blue) indicate pixels to be increased/reduced in intensity to attain higher prediction confidence. At higher CO scores, negative values emerge.

them. (1) It is not clear why some features are rife with artifact-looking patterns, but we can make several hypothesis. The problem might lie with the internal structures of the DNN: the heatmaps in fig. 4(B-E) indicate that we are able to optimize predictive probability with seemingly nonsensical patterns and relative intensity manipulation. This is indicative of the instability of high dimensional neuron activations in a DNN, a sign of fragility against adversarial attack we previously mentioned. How should we address this? In the future, the ideal DNN models have internal structures that work well with GAX, i.e. not only does GAX improve the predictive performance, but heatmaps generated by GAX must correspond to humanly interpretable images, for example, heatmaps that resemble localization-mask or highlight of relevant parts (depending on the explanation need). (2) The regularity of data distribution is probably an important deciding factor in model training. In cases where the X-Ray image are not taken in a regular posture, the empty space can become a false “distinct feature”, as shown in fig. 5. While this may indicate a worrying trend in the flawed training of DNN, and hence a misguided application, we believe GAX can be used to detect such issue before deployments. Related future studies may be aimed at quantifying the effect of skewed distribution on the appearance of such “false prediction” cases. Also see appendix for more, e.g. implementation-specific limitations etc.

Figure 5. An instance of bacterial pneumonia chest X-Ray showing an irregular posture with heavy noise (top right). The empty space might have been used as a false distinct feature for pneumonia classification.

5. Conclusion

We have defined the CO score to measure the effect of using heatmaps in the AX process to directly improve the confidence in making a correct classification. This metric is first demonstrated on a low dimensional example with tractable formulation. AX process has been used to quantify the performance of heatmaps h created by existing XAI methods. We then introduce the Generative Augmentative Explana-
tion (GAX), an optimization of CO score through direct variation of heatmaps as the parameters. The heatmaps can then be directly used to improve model’s prediction. The target $s_{co} \geq 48$ is set for ImageNet and attainable in all cases we tested, a score which exceeds most of the CO scores of other XAI methods. From the results, we support the notion that computationally relevant features are not necessarily relevant to human. Possible novelties in the paper include (1) our main results on the gap in CO scores may be a new indicator in predictive modelling and (2) distinct class-dependent patterns that emerge on GAX-generated heatmaps; they could provide insight into the model’s behaviour.

Software and Data

All codes are available; see main paper and also see appendix.

Acknowledgements

This research was supported by Alibaba Group Holding Limited, DAMO Academy, Health-AI division under Alibaba-NTU Talent Program. The program is the collaboration between Alibaba and Nanyang Technological University, Singapore.

References

Adadi, A. and Berrada, M. Peeking inside the black-box: A survey on explainable artificial intelligence (xai). IEEE Access, 6:52138–52160, 2018. doi: 10.1109/ACCESS.2018.2870052.

Akhtar, N. and Mian, A. Threat of adversarial attacks on deep learning in computer vision: A survey. IEEE Access, 6:14410–14430, 2018. doi: 10.1109/ACCESS.2018.2807385.

Bach, S., Binder, A., Montavon, G., Klauschen, F., Müller, K.-R., and Samek, W. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. PLOS ONE, 10(7):1–46, 07 2015. doi: 10.1371/journal.pone.0130140. URL https://doi.org/10.1371/journal.pone.0130140.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In 2009 IEEE conference on computer vision and pattern recognition, pp. 248–255. Ieee, 2009. URL https://www.kaggle.com/c/imagenet-object-localization-challenge.

Došišić, F. K., Brčić, M., and Hlupić, N. Explainable artificial intelligence: A survey. In 2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), pp. 0210–0215, 2018. doi: 10.23919/MIPRO.2018.8400040.

Fong, R. C. and Vedaldi, A. Interpretable explanations of black boxes by meaningful perturbation. In 2017 IEEE International Conference on Computer Vision (ICCV), pp. 3449–3457, 2017. doi: 10.1109/ICCV.2017.371.

Gilpin, L. H., Bau, D., Yuan, B. Z., Baijsa, A., Specter, M., and Kagal, L. Explaining explanations: An overview of interpretability of machine learning. In 2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA), pp. 80–89, 2018. doi: 10.1109/DSAA.2018.00018.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770–778, 2016.

Krizhevsky, A. One weird trick for parallelizing convolutional neural networks. ArXiv, abs/1404.5997, 2014.

Lakshmanan, L. Why you need to explain machine learning models, Jun 2021. URL https://cloud.google.com/blog/products/ai-machine-learning/why-you-need-to-explain-machine-learning-models.

Lundberg, S. M. and Lee, S.-I. A unified approach to interpreting model predictions. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (eds.), Advances in Neural Information Processing Systems 30, pp. 4765–4774. Curran Associates, Inc., 2017. URL http://papers.nips.cc/paper/7062-a-unified-approach-to-interpreting-model-predictions.pdf.

Mooney, P. Chest x-ray images (pneumonia), Mar 2018. URL https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia.

Olah, C., Mordvintsev, A., and Schubert, L. Feature visualization. Distill, 2(11), November 2017. doi: 10.23915/distill.00007. URL https://doi.org/10.23915%2Fdistill.00007.

Olah, C., Satyanarayan, A., Johnson, I., Carter, S., Schubert, L., Ye, K., and Mordvintsev, A. The building blocks of interpretability, Jan 2020. URL https://distill.pub/2018/building-blocks.

Pesenti, J. Facebook’s five pillars of responsible ai, Jun 2021. URL https://ai.facebook.com/blog/facebook-fives-pillars-of-responsible-ai/.
Augmentative eXplanation and the Distributional Gap of Confidence Optimization Score

Rebuffi, S. A., Fong, R., Ji, X., and Vedaldi, A. There and back again: Revisiting backpropagation saliency methods. In 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 8836–8845, 2020. doi: 10.1109/CVPR42600.2020.00886.

Ribeiro, M. T., Singh, S., and Guestrin, C. “why should i trust you?”: Explaining the predictions of any classifier. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD ’16, pp. 1135–1144, New York, NY, USA, 2016. Association for Computing Machinery. ISBN 9781450342322. doi: 10.1145/2939672.2939778. URL https://doi.org/10.1145/2939672.2939778.

Rudin, C. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. Nature Machine Intelligence, 1(5):206–215, May 2019. ISSN 2522-5839. doi: 10.1038/s42256-019-0048-x. URL https://doi.org/10.1038/s42256-019-0048-x.

Samek, W., Binder, A., Montavon, G., Lapuschkin, S., and Müller, K. Evaluating the visualization of what a deep neural network has learned. IEEE Transactions on Neural Networks and Learning Systems, 28(11):2660–2673, 2017.

Selvaraju, R. R., Das, A., Vedantam, R., Cogswell, M., Parikh, D., and Batra, D. Grad-cam: Why did you say that? visual explanations from deep networks via gradient-based localization. CoRR, abs/1610.02391, 2016. URL http://arxiv.org/abs/1610.02391.

Shrikumar, A., Greenside, P., Shcherbina, A., and Kundaje, A. Not just a black box: Learning important features through propagating activation differences. ArXiv, abs/1605.01713, 2016.

Shrikumar, A., Greenside, P., and Kundaje, A. Learning important features through propagating activation differences. volume 70 of Proceedings of Machine Learning Research, pp. 3145–3153, International Convention Centre, Sydney, Australia, 06–11 Aug 2017. PMLR. URL http://proceedings.mlr.press/v70/shrikumar17a.html.

Simonyan, K., Vedaldi, A., and Zisserman, A. Deep inside convolutional networks: Visualising image classification models and saliency maps. In Workshop at International Conference on Learning Representations, 2014.

Springenberg, J. T., Dosovitskiy, A., Brox, T., and Riedmiller, M. A. Striving for simplicity: The all convolutional net. CoRR, abs/1412.6806, 2015.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., and Fergus, R. Intriguing properties of neural networks. CoRR, abs/1312.6199, 2014.

Tjoa, E. and Guan, C. A survey on explainable artificial intelligence (xai): Toward medical xai. IEEE Transactions on Neural Networks and Learning Systems, pp. 1–21, 2020a. doi: 10.1109/TNNLS.2020.3027314.

Tjoa, E. and Guan, C. Quantifying explainability of saliency methods in deep neural networks. ArXiv, abs/2009.02899, 2020b.

Zeiler, M. D. and Fergus, R. Visualizing and understanding convolutional networks. In Fleet, D., Pajdla, T., Schiele, B., and Tuytelaars, T. (eds.), Computer Vision – ECCV 2014, pp. 818–833, Cham, 2014. Springer International Publishing. ISBN 978-3-319-10590-1.

Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., and Torralba, A. Learning deep features for discriminative localization. In 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2921–2929, June 2016. doi: 10.1109/CVPR.2016.319.
A. Appendix

All codes are available (see main text). All instructions to reproduce the results can be found in README.md, given as command line input, such as:

1. python main_pneu.py --mode xai_collect --model resnet34 --PROJECT_ID pneu256n_1 --method Saliency --split train --realtime_print 1 --n_debug 0

2. python main_pneu.py --mode gax --PROJECT_ID pneu256n_1 --model resnet34 --label NORMAL --split test --first_n_correct 100 --target_co 48 --gax_learning_rate 0.1

The whole experiment can be run on small GPU like NVIDIA GeForce GTX 1050 with 4 GB dedicated memory.

The codes are run on Python 3.8.5. The only specialized library used is Pytorch (specifically torch==1.8.1+cu102, torchvision==0.9.1+cu102) and Pytorch Captum (captum==0.3.1). Other libraries are common python libraries.

Regarding Captum. We replace Pytorch Captum “inplace relu” so that some attribution methods will work properly (see adjust_for_captum_problem in model.py where applicable).

We also manually edit non-full backward hooks in the source codes to prevent the gradient propagation issues. For example, from Windows, see Lib \site-packages \captum \attr \core \guided_backprop\deconvnet.py, function def _register_hooks(self, module: Module). There is a need to change from hook = module. register_backward_hook(self.backward_hook) to hook = module. register_full_backward_hook(self.backward_hook).

A.1. More interpretations in low dimensional example

Interpretation of attribute values for non-negative less distinct components. Now, we consider data sample with lower $a_1 = 0.7$ (i.e. less distinct) but components are still non-negative. Fig. 1 middle shows that components are still non-negative around $\theta \in [\pi/8, 3\pi/8]$. Similar attribution of $h_1$ and suppression of $h_2$ are observed similarly although with lower magnitude around $\theta \approx 0$. At $\theta \approx \pi/4$, similar difficulty in distinguishing homogenous transformation is present, naturally. Further rotation to $3\pi/8$ will give higher $h_2$ as well. Fig. 6 right shows similar behavior even for $a_1 \approx a_2$, though non-negative values are observed for rotations around $[-\pi/4, \pi/4]$. The sample is barely categorized as $c = 1$ since $a_1 > a_2$. However, the resulting attribution values still highlights the positive contribution $x_1$, primarily through higher $h_1$ attribution value, even though the magnitudes are lower compared to previous examples.

Interpretation of attribute values for negative components. Beyond the rotation range that yields non-negative components, we do see negative components $x_i < 0$ assigned highly negative $h_i$ values. For example, fig. 6 left at $\theta \approx \pi$ shows a rotation of the components to the negatives. In this formulation, negative attribution values are assigned to negative components naturally, because $w \star x$ starts with $w_i = 1$ and $x_i < 0$, as $w_i$ is optimized, our example shows an instance where, indeed, we need higher $w_i$, very negative $h_1$. Recall the main interpretation. In the end, this high negative attribution is aimed at improving CO score. The large negative $h_1$ component increases the likelihood of predicting $c = 1$; conversely, the relatively low $h_2$ magnitude increases the same likelihood. Therefore, we do not immediately conclude that negative attribution values contribute “negatively” to prediction, which is a term sometimes ambiguously used in XAI community. In practice, case by case review may be necessary.

A.2. Zero CO scores and other scores

Zero CO score might occur when when $h$ yields uniform change to the output, i.e. $DNN(x + h) = DNN(x) + c$ for some constant $c$. This is obtained by simply plugging into the CO score formula. Special case may occur when $h$ is constant over all pixels, especially when $N(g(x + h)) = N(g(x))$ for some intermediate normalization layer $N$ and intermediate pre-normalized composition of layers $g = g_k \circ g_{k-1} \circ \cdots \circ g_1$.

Positive CO score indicates that the component $[s_{ci}]_i$, where $i$ corresponds to the groundtruth label, is increased by $h$ at a greater magnitude than the average of all other components, which in turn means that the component $DNN(x + h)_i$ is similarly increased at greater magnitude compared to the average of other components. Hence, the prediction favours component $i$ relatively more, i.e. the probability of predicting the correct class is increased. Negative CO score is simply the reverse: probability of predicting the correct class has been reduced relatively.
A.3. More Boxplots of CO Scores.

Fig. 7 shows more variations of CO scores in our experiments, similar to the ones shown in the main text. Some scores clearly demonstrate distinct gaps in CO scores between the correct and wrong predictions.

From fig. 8, AX process is applied to the heatmaps generated in different layers of ResNet34. We expect higher improvement of CO scores for AX process using heatmaps from deeper layers that are known to detect more features. We do observe a difference in $x \ast h$ AX process, but not in $x + h$ for Layer Grad CAM.

A.4. More Considerations, Limitations and Future Works

Different GAX and empirical choices in implementation. Parts of the implementations, such as the initialization of $w$ to 1.0, are nearly arbitrary, though it is the first choice made from the 2D example that happens to work. Different implementations come with various trade-offs. Most notably, the choice of learning rate $0.1$ is manually chosen for its reliable and fast convergence, although convergence is attainable for smaller learning rate like $0.001$ after longer iterations. However, we need to include more practical considerations. For example, saving heatmaps iteration by iteration will generally consume around 5-12 MB of memory for current choices. Longer optimization iterations may quickly cause a blow-up, and there is no known fixed number of iterations needed to achieve convergence to the target CO score. Saving heatmaps at certain CO scores milestones can be considered, though we might miss out on important heatmap changes in between.

Parameter selection process is thus not straightforward. For practical purposes, learning rates can be tested in order of ten, $10^n$, and other parameters can be tested until a choice is found where each optimization process converges at a rate fast enough for nearly instantaneous, quick diagnosis. Other choices of optimizers with different parameters combination can be explored as well, though we have yet to see dramatic changes.

GAX, different DNN architectures and different datasets. Comparisons are tricky, since different architectures might behave differently at their FC end. For example, for Saliency method on ImageNet, Alexnet’s boxplot of CO scores in appendix fig. 7(A) right ($x \ast h \ast w \ast x$) shows a wider range of CO scores than that of Resnet34 in fig. ?? . Comparison of CO scores on Chest X-Ray dataset shows even larger variability. Furthermore, recall that we illustrated using the 2D example the reason we avoid sigmoid function: suppressed change in the CO score due to its asymptotic part. We have avoided Softmax for similar reason, and a further study can be conducted to characterize the scores with Softmax or other modifications. From here, the ideal vision is to develop a model that scales with CO score in not only a computationally relevant way, but also in a human relevant way: we want a model that increases predictive probability when the heatmap highlights exactly the correct localization of the objects or highly relevant features related to the objects. This is a tall effort, particularly because explanations are highly context dependent. Transparent and trustworthy applications of DNN may benefit from the combined improvements in humanly understandable context and computationally relevant attributions built around that context.
Figure 7. Boxplots of CO scores for existing XAI methods, including another GAX implementation $x \ast h = x \ast (w \ast x)$.

Figure 8. Boxplots of CO scores for heatmaps from Layer GradCAM for ResNet34 and ImageNet dataset. CO scores of heatmaps generated from different layers (and resized accordingly) are shown.