HINT: Hierarchical Neuron Concept Explainer

Andong Wang, Wei-Ning Lee, Xiaojuan Qi
The University of Hong Kong
wangad@connect.hku.hk, wnlee@eee.hku.hk, xjqi@eee.hku.hk

Abstract

*To interpret deep networks, one main approach is to associate neurons with human-understandable concepts. However, existing methods often ignore the inherent connections of different concepts (*e.g.*, dog and cat both belong to animals), and thus lose the chance to explain neurons responsible for higher-level concepts (*e.g.*, animal). In this paper, we study hierarchical concepts inspired by the hierarchical cognition process of human beings. To this end, we propose HIerarchical Neuron concepT explainer (**HINT**) to effectively build bidirectional associations between neurons and hierarchical concepts in a low-cost and scalable manner. HINT enables us to systematically and quantitatively study whether and how the implicit hierarchical relationships of concepts are embedded into neurons. Specifically, HINT identifies collaborative neurons responsible for one concept and multimodal neurons pertinent to different concepts, at different semantic levels from concrete concepts (*e.g.*, dog) to more abstract ones (*e.g.*, animal). Finally, we verify the faithfulness of the associations using Weakly Supervised Object Localization, and demonstrate its applicability in various tasks, such as discovering saliency regions and explaining adversarial attacks. Code is available on* https://github.com/AntonotnaWang/HINT.

1. Introduction

Deep neural networks have attained remarkable success in many computer vision and machine learning tasks. However, it is still challenging to interpret the hidden neurons in a human-understandable manner, which is of great significance in uncovering the reasoning process of deep networks and increasing the trustworthiness of deep learning to humans [3, 31, 61].

Early research focuses on finding evidence from input data to explain deep model predictions [4, 10, 29, 33, 34, 48, 51, 52, 54–57, 64], where the neurons remain unexplained. More recent efforts have attempted to associate hidden neurons with human-understandable concepts [7–9, 11, 23, 44, 45, 67, 68, 71, 72]. Although insightful interpretations of neurons' semantics have been demonstrated, *i.e.*, identification of the neurons controlling contents of *trees* [8], existing methods define the concepts in an ad-hoc manner which heavily relies on human annotations, such as manual visual inspection [11, 44, 45, 72], manually labeled classification categories [23], or hand-crafted guidance images [7–9, 71]. They thus suffer from heavy costs and scalability issues. Moreover, existing methods often ignore the inherent connections among different concepts (*e.g.*, *dog* and *cat* both belong to *mammal*), and treat them independently, which therefore loses the chance to discover neurons responsible for implicit higher-level concepts (*e.g.*, *canine*, *mammal*, and *animal*) and explore whether the network can create abstractions of things like our humans do.

The above motivates us to rethink how concepts should be defined to more faithfully reveal the roles of hidden neurons. We draw inspirations from the hierarchical cognition process of human beings– human tend to organize things from specific to general categories [37, 47, 60]– and propose to explore hierarchical concepts which can be harvested from WordNet [39] (a lexical database of semantic relations between words). We investigate whether deep networks can automatically learn the hierarchical relationships of categories that were not labeled in the training data. More concretely, we aim to identify neurons for both low-level concepts such as *Malamute*, *Husky*, and *Persian cat*, and the implicit higher-level concepts such as *dog* and *animal* as shown in Figure 1 (a). Note that we call less abstract concepts low-level and more abstract concepts high-level.

To this end, we develop **HIerarchical Neuron concepT explainer** (**HINT**), which builds a bidirectional association between neurons and hierarchical concepts (see Figure 1). First, we develop a saliency-guided approach to identify the high dimensional representations associated with the hierarchical concepts on hidden layers (noted as responsible regions in Figure 1 (b)), making HINT low-cost and scalable as no extra hand-crafted guidance is required. Then, we train classifiers shown in Figure 1 (c) to separate different concepts' responsible regions, where the weights represent the contribution of the corresponding neuron to the classification. Based on the classifiers, we design a Shap-

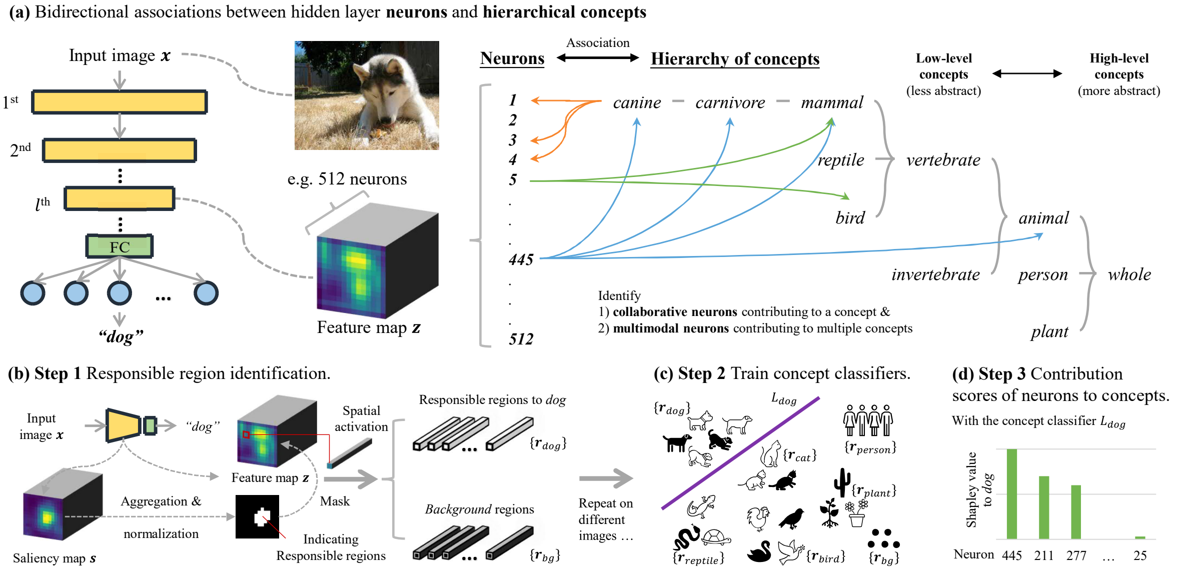
(a) Bidirectional associations between hidden layer neurons and hierarchical concepts
Input image x
1st
2nd
lth
FC
"dog"
e.g. 512 neurons
Feature map z
Neurons
Association
Hierarchy of concepts
Low-level concepts (less abstract)
High-level concepts (more abstract)
1
2
3
4
5
445
512
canine
carnivore
mammal
reptile
bird
vertebrate
invertebrate
animal
person
plant
whole
Identify
1) collaborative neurons contributing to a concept &
2) multimodal neurons contributing to multiple concepts
(b) Step 1 Responsible region identification.
Input image x
"dog"
Feature map z
Spatial activation
Saliency map s
Aggregation & normalization
Mask
Indicating Responsible regions
Responsible regions to dog
$\{r_{dog}\}$
Background regions
$\{r_{bg}\}$
Repeat on different images ...
(c) Step 2 Train concept classifiers.
$\{r_{dog}\}$
L_{dog}
$\{r_{cat}\}$
$\{r_{person}\}$
$\{r_{plant}\}$
$\{r_{reptile}\}$
$\{r_{bird}\}$
$\{r_{bg}\}$
(d) Step 3 Contribution scores of neurons to concepts.
With the concept classifier L_{dog}
Shapley value to dog
Neuron 445 211 277 ... 25

Figure 1. Overall illustration of HINT. **(a)** HINT is able to build bidirectional associations between hidden layer neurons and hierarchical concepts. It can also identify collaborative neurons and multimodal neurons. Further, HINT helps to indicate how the neurons learn the hierarchical relationships of categories. **(b)-(c)** Main steps. See Section 3.1 for Step 1, Section 3.2 for Step 2, and Section 3.3 for Step 3.

ley value-based scoring method to fairly evaluate neurons' contributions, considering both neurons' individual and collaborative effects.

To our knowledge, HINT presents the first attempt to associate neurons with hierarchical concepts, which enables us to systematically and quantitatively study whether and how hierarchical concepts are embedded into deep network neurons. HINT identifies collaborative neurons contributing to one concept and multimodal neurons contributing to multiple concepts. Especially, HINT finds that, despite being trained with only low-level labels, such as *Husky* and *Persian cat*, deep neural networks automatically embed hierarchical concepts into its neurons. Also, HINT is able to discover responsible neurons to both higher-level concepts, such as *animal*, *person* and *plant*, and lower-level concepts, such as *mammal*, *reptile* and *bird*.

Finally, we verify the faithfulness of neuron-concept associations identified by HINT with a Weakly Supervised Object Localization task. In addition, HINT achieves remarkable performance in a variety of applications, including saliency method evaluation, adversarial attack explanation, and COVID19 classification model evaluation, further manifesting the usefulness of HINT.

2. Related Work

Neuron-concept Association Methods. Neuron-concept association methods aim at directly interpreting the internal computation of CNNs [2, 12, 25, 43]. Early works show that neurons on shallower layers tend to learn simpler concepts, such as lines and curves, while higher layers tend to learn more abstract ones, such as heads or legs [63, 64]. TCAV [30] and related studies [22, 24] quantify the contribution of a given concepts represented by guidance images to a target class on a chosen hidden layer. Object Detector [72] visualizes the concept-responsible region of a neuron in the input image by iteratively simplifying the image. After that, Network Dissection [7, 8, 71] quantifies the roles of neurons by assigning each neuron to a concept with the guidance of extra images. GAN Dissection [8, 9] illustrates the effect of concept-specific neurons by altering them and observing the emergence and vanishing of concept-related contents in images. Neuron Shapley [23] identifies the most influential neuron over all hidden layers to an image category by sorting Shapley values [49]. Besides pre-defined concepts, feature visualization methods [11, 44, 45] generate Deep Dream-style [42] explanations for each neuron and manually interpret their meanings. Additionally, Net2Vec [20] maps semantic concepts to vectorial embeddings to investigate the relationship between CNN filters and concepts. However, existing methods cannot systematically explain how the network learns the inherent connections of concepts and suffer from high cost and scalability issues. HINT is proposed to overcome these limitations and goes beyond exploring each concept individually. Specifically, HINT adopts hierarchical concepts to explore their semantic relationships.

Saliency Map Methods. Saliency map methods are a stream of simple and fast interpretation methods which show the pixel responsibility (*i.e.* saliency score) in the input image for a target model output. There are two main categories of saliency map methods – backpropagation-based and perturbation-based. Backpropagation-based methods mainly generate saliency maps by gradients; they include Gradient [52], Gradient x Input [51], Guided Backpropagation [55], Integrated Gradient [57], SmoothGrad [54], LRP [5, 26], Deep Taylor [41], DeepLIFT [50], and Deep SHAP [13]. Perturbation-based saliency methods perturbate input image pixels and observe the variations of model outputs; they include Occlusion [64], RISE [46], Real-time [15], Meaningful Perturbation [21], and Extremal Perturbation [19]. Inspired by saliency methods, in HINT, we build a saliency-guided approach to identify the responsible regions for each concept on hidden layers.

3. Method

Overview. Considering a CNN classification model f and a hierarchy of concepts $\mathcal{E} : \{e\}$ (see Figure 1 (a)), our goal is to identify bidirectional associations between neurons and hierarchical concepts. To this end, we develop **HIerarchical Neuron concepT explainer** (**HINT**) to quantify the contribution of each neuron d to each concept e by a contribution score ϕ where higher contribution value means a stronger association between d and e, and vice versa.

The key problem therefore becomes how to estimate the score ϕ for any pair of e and d. We achieve this by identifying how the network maps concept e to a high dimensional space and quantifies the contribution of d for the mapping. First, given a concept e and an image $\boldsymbol{x}$, on feature map $\boldsymbol{z}$ of the l^{th} layer, HINT identifies the responsible regions $\boldsymbol{r}_e$ to concept e by developing a saliency-guided approach elaborated in Section 3.1. Then, given the identified regions for all the concepts, HINT further trains concept classifier L_e to separate concept e's responsible regions $\boldsymbol{r}_e$ from other regions $\boldsymbol{r}_{\mathcal{E}\setminus e} \cup \boldsymbol{r}_{b^*}$, where b^* represents background (see Section 3.2). Finally, to obtain ϕ, we design a Shapley value-based approach to fairly evaluate the contribution of each neuron d from the concept classifiers (see Section 3.3).

3.1. Responsible Region Identification for Concepts

In this section, we introduce our saliency-guided approach to collect the responsible regions $\boldsymbol{r}_e$ for a certain concept $e \in \mathcal{E}$ to serve as the training samples of the concept classifier which will be described in Section 3.2.

Taking an image $\boldsymbol{x}$ containing a concept e as input, the network f generates a feature map $\boldsymbol{z} \in \mathbb{R}^{D_l \times H_l \times W_l}$ where there are D_l neurons in total. Generally, not all regions of $\boldsymbol{z}$ are equally related to e [68]. In other words, some regions have stronger correlations with e while others are less correlated, as shown in Figure 1 (b) "Step 1". Based on the above observation, we propose a saliency-guided approach to identify the closely related regions $\boldsymbol{r}_e$ to the concept e in feature map z. We call them responsible regions.

Algorithm 1: HINT

```
Input: A set of images with hierarchical concepts
       $\{(\boldsymbol{x}, e)\}$, a set of neurons $\mathcal{D}$ for experiment,
       modified saliency method $\Lambda$, aggregation
       method $\zeta$, and threshold $t \in (0, 10$.
Output: Score matrix $\Phi$ where every element $\phi$ is
        the Shapely value of neuron $d$ to concept $e$.
Init: Responsible region containers $\boldsymbol{r}_e = \{\ \}$ for
      each $e$ in $\mathcal{E}$, background region container
      $\boldsymbol{r}_{b^*} = \{\ \}$, and score matrix $\Phi = \{0\}^{|\mathcal{D}| \times |\mathcal{E}|}$.
for each $(\boldsymbol{x}, e)$ do
    feature map $\boldsymbol{z} = f_l(\boldsymbol{x})$ ;
    saliency map $\boldsymbol{s} = \Lambda(\boldsymbol{x}, f_l \mid e)$ ;
    $\boldsymbol{z} \leftarrow \boldsymbol{z}_{\mathcal{D},:,:}$ ;
    $\boldsymbol{s} \leftarrow \boldsymbol{s}_{\mathcal{D},:,:}$ ;
    $\hat{\boldsymbol{s}} = Normalization(\zeta(\boldsymbol{s})) \in [0,1]^{H_l \times W_l}$ ;
    $\boldsymbol{z}_e = \boldsymbol{z} \odot (\hat{\boldsymbol{s}} \geq t)$, add $\boldsymbol{z}_e$ to $\boldsymbol{r}_e$ ;
    $\boldsymbol{z}_{b^*} = \boldsymbol{z} \odot (\hat{\boldsymbol{s}} < t)$, add $\boldsymbol{z}_{b^*}$ to $\boldsymbol{r}_{b^*}$ ;
for each $e$ in $\mathcal{E}$ do
    Train classifier $L_e$ which separates $\boldsymbol{r}_e$ and
      $\boldsymbol{r}_{\mathcal{E}\setminus e} \cup \boldsymbol{r}_{b^*}$
for each $e$ in $\mathcal{E}$ do
    for each $d$ in $\mathcal{D}$ do
        $\phi =$ Shapley value of neuron $d$ to concept $e$;
        Update $\Phi$ with $\phi$;
```

First, we obtain the saliency map on the l^{th} layer. As shown in Figure 1 (b) "Step 1", with the feature map $\boldsymbol{z}$ on the l^{th} layer extracted, we derive the l^{th} layer's saliency map $\boldsymbol{s}$ with respect to concept e by the saliency map estimation approach Λ. Note that HINT is compatible with different back-propagation based saliency map estimation methods. We implement five of them [51, 52, 54, 55, 57], please refer to the Supplementary Material for more details. Note that different from existing works [51, 52, 54, 55, 57] that pass the gradients to the input image as saliency scores, we early stop the back-propagation at the l^{th} layer to obtain the saliency map $\boldsymbol{s}$. Here, we use modified SmoothGrad [54] as an example to demonstrate our approach: $\Lambda = \frac{1}{N} \sum_{n=1}^{N} \frac{\partial f^e(\boldsymbol{x}')}{\partial \boldsymbol{z}'}$ where $\boldsymbol{x}' = \boldsymbol{x} + \mathcal{N}(\mu, \sigma_n^2)$ and $\mathcal{N}$ indicates normal distribution. It is notable that we may also optimally select part of the neurons $\mathcal{D}$ for analysis.

Next is to identify the responsible regions on feature map $\boldsymbol{z}$ with the guidance of the saliency map $\boldsymbol{s}$. Specifically, we categorize each entry $z_{\mathcal{D},i,j}$ in $\boldsymbol{z}$ to be responsible to e or not. To this end, the saliency map $\boldsymbol{s}$ is first aggregated by an aggregation function ζ along the channel dimension and then normalized to be within $[0, 1]$. Note that different aggregation functions ζ can be applied (see five different ζ shown in Supplementary Material). Here, we aggregate

$\boldsymbol{s}$ using Euclidean norm $\zeta = \|\boldsymbol{s}\|$ along its first dimension. After that, we obtain $\hat{\boldsymbol{s}} \in [0,1]^{H_l \times W_l}$ with each element $s_{i,j}$ indicating the relevance of $z_{\mathcal{D},i,j}$ to concept e. By setting a threshold t (we set t as 0.5 in the paper) and masking $\boldsymbol{z}$ with $\hat{\boldsymbol{s}} \geq t$ and $\hat{\boldsymbol{s}} < t$, we finally obtain responsible regions and background regions respectively (see the illustration of the two regions Figure 1 (b): "Step 1").

Our saliency-guided approach extends the interpretability of saliency methods, which originally aim to find the "responsible regions" to a concept on one particular image. However, our approach is able to identify "responsible regions" to a concept on the high dimensional space of a hidden layer from multiple images, which can more accurately describe how the network represents concept e internally. Therefore, our saliency-guided approach provides better interpretability as it helps us to investigate the internal abstraction of concept e in the network.

3.2. Training of Concept Classifiers

For all images, we identify its responsible regions for each concept $e \in \mathcal{E}$ following the procedures described in 3.1 and construct a dataset which contains a collection of responsible regions $\boldsymbol{r}_e$ and a collection of background regions $\boldsymbol{r}_{b^*}$. Given the dataset, as shown in Figure 1 (c) "Step 2", we use the high dimensional CNN hidden layer features to train a concept classifier L_e which distinguishes $\boldsymbol{r}_e$ from $\boldsymbol{r}_{\mathcal{E}\backslash e} \cup \boldsymbol{r}_{b^*}$, *i.e.*, separating concept e from other concepts $\mathcal{E}\backslash e \cup b^*$ (Line 9 and 10 in Algorithm 1).

L_e can have many forms: a linear classifier, a decision tree, a Gaussian Mixture Model, and so on. Here, we use the simplest form, a linear classifier, which is equivalent to a hyperplane separating concept e from others in the high dimensional feature space of CNN.

$$L_e(r) = \sigma\left(\boldsymbol{\alpha}^T r\right), \qquad (1)$$

where $r = z_{\mathcal{D},i,j} \in \mathbb{R}^{|\mathcal{D}|}$ represents spatial activation with each element representing a neuron; $\boldsymbol{\alpha}$ is a vector of weights, σ is a sigmoid function, and $L_e(r) \in [0,1]$ represents the confidence of r related to a concept e.

It is notable that we can apply the concept classifier L_e back to the feature map $\boldsymbol{z}$ to visualize how L_e detect concept e. Classifiers of more abstract concepts (*e.g.*, *whole*) tend to activate regions of more general features, which helps us to locate the entire extent of the object. On the contrary, classifiers of lower-level concepts tend to activate regions of discriminative features, such as eyes and heads.

3.3. Contribution Scores of Neurons to Concepts

Next is to decode the contribution score ϕ from the concept classifiers. A simple method to estimate ϕ is to use the learned classifier weights corresponding to each neuron e, where a higher value typically means a larger contribution [40]. However, the assumption that α can serve as the contribution score is that the neurons are independent of each other. However, it is generally not true. To achieve a fair evaluation of neurons' contributions to e, a Shapley value-based approach is designed to calculate the scores ϕ, which considers neurons' individual effects as well as the contributions coming from the collaboration with others.

Shapely value [49] is from Game Theory, which evaluates channels' individual and collaborative effects. More specifically, if a channel cannot be used for classification independently but can greatly improve classification accuracy when collaborating with other channels, its Shapley value can still be high. Shapely value satisfies the properties of efficiency, symmetry, dummy, and additivity [40]. Monte-Carlo sampling is used to estimate the Shapley values by testing the target neuron's possible coalitions with other neurons. Equation (2) shows how we calculate Shapley value ϕ of a neuron d to concept e.

$$\phi = \frac{\sum_{\boldsymbol{r}} \left| \sum_{i=1}^{M} \left(L_e^{\langle \mathcal{S} \cup d \rangle}(r) - L_e^{\langle \mathcal{S} \rangle}(r) \right) \right|}{M |\boldsymbol{r}_{\mathcal{E}} \cup \boldsymbol{r}_{b^*}|}, \qquad (2)$$

where $r = z_{\mathcal{D},i,j}$ represents spatial activation from $\boldsymbol{r}_{\mathcal{E}}$ and $\boldsymbol{r}_{b^*}$; $\mathcal{S} \subseteq \mathcal{D}\backslash d$ is the neuron subset randomly selected at each iteration; $\langle * \rangle$ is an operator keeping the neurons in the brackets, *i.e.*, $\mathcal{S} \cup d$ or $\mathcal{S}$, unchanged while randomizing others; M is the number of iterations of Monte-Carlo sampling; $L_e^{\langle * \rangle}$ means that the classifier is re-trained with neurons in the brackets unchanged and others being randomized.

By repeating the calculation for different e and d (see Line 11 to line 14 in Algorithm 1), finally, we can get the score matrix Φ.

3.4. Neuron-Concept Association

By repeating the score calculations for all pairs of e and d, we obtain a score matrix Φ, where each row represents a neuron d and each column represents a concept e in the hierarchy. By sorting the scores in the column of concept e, we can get collaborative neurons all having high contributions to a concept e. Also, by sorting the scores in the row of neuron d, we can test whether d is multimodal (having high scores to multiple concepts) and observe a hierarchy of concepts that d is responsible for.

Note that the score matrix Φ cannot tell us the exact number of responsible neurons to concept e. For a contribution score ϕ which is zero or near zero, the corresponding neuron d can be regarded as irrelevant to the corresponding concept e. Therefore, for truncation, we may set a threshold for ϕ. In our experiment, for a concept, we sort scores and select the top N as responsible neurons.

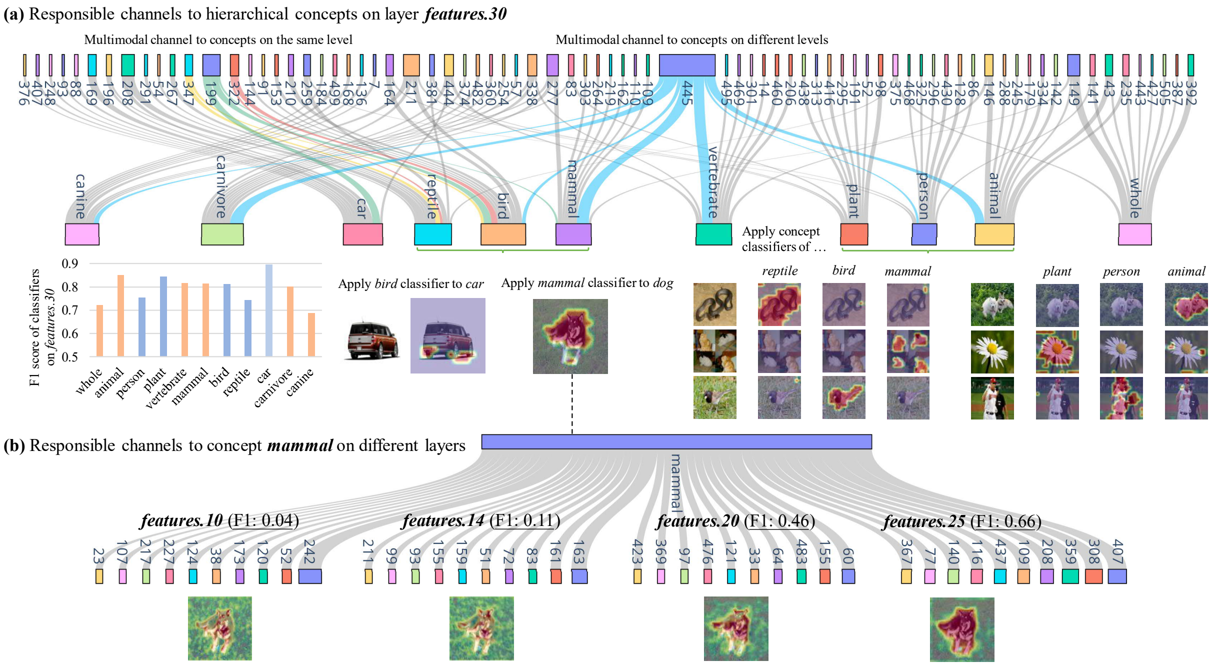

Figure 2. Bidirectional associations between neurons and hierarchical concepts. The width of the link indicates the size of the contribution score of a neuron to a concept. **(a)** Responsible neurons to hierarchical concepts (see the hierarchy in Figure 1) on layer features.30 in VGG19. The F1 scores of concept classifiers show their capability of distinguishing the target concepts. The pictures illustrate the results of applying concept classifiers on different images. For most of the cases, the concept classifiers only locate the objects belonging to the target concepts. However, as *bird* and *car* share multimodal neurons, the *bird* classifier responses to the wheels of the car. **(b)** Responsible neurons to *mammal* on different layers. The pictures and F1 scores indicate the network can more easily distinguish *mammal* from other concepts as the layer goes higher.

4. Experiments

4.1. Experimental setup

HINT is a general framework which can be applied on any CNN architectures. We evaluate HINT on several models trained on ImageNet [17] with representative CNN backbones including VGG-16 [53], VGG-19 [53], ResNet-50 [27], and Inception-v3 [58]. In this paper, the layer names are from PyTorch pretrained models (*e.g.*, "features.30" is a layer name of VGG19). The hierarchical concept set $\mathcal{E}$ is built upon the 1000 categories of ImageNet with hierarchical relationship is defined by WordNet [39] as shown in Figure 1. Figure 3 shows the computational complexity analysis, indicating that Shapely value calculation is negligible when considering the whole cycle.

4.2. Responsible Neurons to Hierarchical Concepts

In this section, we study the responsible neurons for the concepts and show the hierarchical cognitive pattern of CNNs. We adopt the VGG-19 backbone and show the top-10 significant neurons to each concept (N=10). The results in Figure 2 manifest that HINT explicitly reveals the hierarchical learning pattern of the network. Some neurons are responsible for concepts with higher semantic levels, such as *whole* and *animal*, and others are for more detailed concepts, such as *canine*. Besides, HINT shows that there can be multiple neurons contributing to a single concept, and HINT also identifies multimodal neurons, which have high contributions to multiple concepts.

Concepts of different levels. First, we investigate the concepts of different levels in Figure 2 (a). Among all the concepts, *whole* has the highest semantic level, including *animal*, *person*, and *plant*. To study how a network recognizes a *Husky* (a subclass of *canine*) image on a given layer, HINT hierarchically identifies the neurons which are responsible for the concept from higher levels (like *whole*, *animal*) to lower ones (like *canine*). Besides, HINT is able to identify multimodal neurons which take responsibility to many concepts at different semantic levels. For example, the 445^{th} neuron delivers the most contribution to multiple concepts, including *animal*, *vertebrate*, *mammal*, and *carnivore*, and also contributes to *canine*, manifesting that the 445^{th} neuron captures the general and species-specific features which are not labeled in the training data.

Concepts of the same level. Next, we study the responsible neurons for concepts at the same level identified by

HINT. For *mamml*, *reptile*, and *bird*, there exist multimodal neurons as the three categories share morphological similarities. For example, the 199^{th} and 445^{th} neurons contribute to both *mammal* and *bird*, while the 322^{nd} and 347^{th} neurons are individually responsible for both *reptile* and *bird*. Interestingly, HINT identifies multimodal neurons contributing to concepts which are conceptually far part to humans. For example, the 199^{th} neuron contributes to both *bird* and *car*. By applying the *bird* classifier to images of *bird* and *car*, we find that the body of the *bird* and the wheels of the *car* can be both detected.

Same concept on different layers. We also identify responsible neurons on different network layers with HINT. Figure 2 (b) illustrates the 10 most responsible neurons to *mammal* in other four network layers. On shallow layers, such as on layer features.10, HINT indicates that the concept of *mammal* cannot be recognized by the network (F1 score: 0.04). However, as the network goes deeper, the F1 score of *mammal* classifier increases until around 0.8 on layer features.30, which is consistent with the existing works [63, 64] that deeper layers capture higher-level and richer semantic meaningful features.

4.3. Verification of Associations by Weakly Supervised Object Localization

With the associations between neurons and hierarchical concepts obtained by HINT, we further validate the associations using Weakly Supervised Object Localization (WSOL). Specifically, we train a concept classifier L_e (see detailed steps in Section 3.1 and 3.2) with the top-N significant neurons corresponding to concept e at a certain layer, and locate the responsible regions using L_e as the localization results. Good localization performance of L_e indicates the N neurons also have high contributions to concept e.

Comparison of localization accuracy. Quantitative evaluation in Table 1 and 2 show that HINT achieves comparable performance to existing WSOL approaches, thus validating the associations. We train *animal* (Table 1) and *whole* (Table 2) classifiers with 10%, 20%, 40%, 80% neurons sorted and selected by Shapley values on layer "features.26" (512 neurons) of VGG16, layer "layer3.5" (1024 neurons) of ResNet50, and layer "Mixed_6b" (768 neurons) of Inception v3, respectively. To be consistent with the commonly-used WSOL metric, Localization Accuracy measures the ratio of images with IoU of groundtruth and predicted bounding boxes larger than 50%. In Table 1, we compare HINT with the state-of-the-art methods on dataset CUB-200-2011 [59], which contains images of 200 categories of birds. Note that existing localization methods need to re-train the model on the CUB-200-2011 as they are tailored to the classifier while HINT directly adopts the classifier trained on ImageNet without further finetuning on CUB-200-2011. Even so, HINT still achieves a comparable performance when adopting VGG16 and Inception v3, and performs the best when adopting ResNet50. However, Table 2 shows that HINT outperforms all existing methods on all models on ImageNet. Besides, the differences in Localization Accuracy may indicate different models have different learning modes. Precisely, few neurons in VGG16 are responsible for *animal* or *whole*, while most neurons in ResNet50 contribute to identifying *animal* or *whole*. In conclusion, the results quantitatively prove that the associations are valid, and HINT achieves comparable performance to WSOL. More analysis is included in the supplementary file.

Table 1. Comparison of Localization Accuracy on CUB-200-2011. * indicates fine-tuning on CUB-200-2011.

	VGG16	ResNet50	Inception v3
CAM* [73]	34.4%	42.7%	43.7%
ACoL* [69]	45.9%	-	-
SPG* [70]	-	-	46.6%
ADL* [14]	52.4%	62.3%	53.0%
DANet* [62]	52.5%	-	49.5%
EIL* [36]	57.5%	-	-
PSOL* [65]	66.3%	70.7%	65.5%
GCNet* [32]	63.2%	-	-
RCAM* [6]	59.0%	59.5%	-
FAM* [38]	**69.3%**	**73.7%**	**70.7%**
Ours (10%)	**66.6%**	60.2%	49.0%
Ours (20%)	65.2%	67.1%	55.8%
Ours (40%)	61.3%	77.3%	52.8%
Ours (80%)	64.8%	**80.2%**	**56.2%**

Table 2. Comparison of Localization Accuracy on ImageNet.

	VGG16	ResNet50	Inception v3
CAM [73]	42.8%	-	-
ACoL [69]	45.8%	-	-
SPG [70]	-	-	48.6%
ADL [14]	44.9%	48.5%	48.7%
DANet [62]	-	-	48.7%
EIL [36]	46.8%	-	-
PSOL [65]	50.9%	54.0%	54.8%
GCNet [32]	-	-	49.1%
RCAM [6]	44.6%	49.4%	-
FAM [38]	**52.0%**	**54.5%**	**55.2%**
Ours (10%)	64.7%	59.7%	53.1%
Ours (20%)	**66.1%**	66.6%	54.1%
Ours (40%)	64.4%	69.4%	54.3%
Ours (80%)	62.6%	**70.7%**	**58.7%**

Flexible choice of localization targets. When locating objects, HINT has a unique advantage: a flexible choice of localization targets. We can locate objects on different levels in the concept hierarchy (*e.g.*, *bird*, *mammal*, and *animal*). In experiments, we train concept classifiers of *whole*, *person*, *animal*, and *bird* using 20 most important neurons on layer features.30 of VGG19 and apply them on PASCAL VOC 2007 [18]. Figure 4 (a) shows that HINT can accurately locate the objects belonging to different concepts.

Extension to locate the entire extent of the object. Many

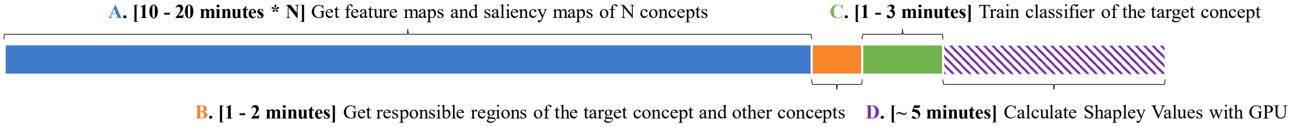
A. [10 - 20 minutes * N] Get feature maps and saliency maps of N concepts
C. [1 - 3 minutes] Train classifier of the target concept
B. [1 - 2 minutes] Get responsible regions of the target concept and other concepts
D. [~ 5 minutes] Calculate Shapley Values with GPU

Figure 3. Time consumption for different stages of HINT. The most time consuming part is the data preparation process. Shapely value computation takes about 5 minutes with a single NVIDIA RTX 2080, while linear classifier training takes $1 - 3$ minutes. Therefore, the time consumption of Shapely value calculation is negligible when considering the whole cycle.

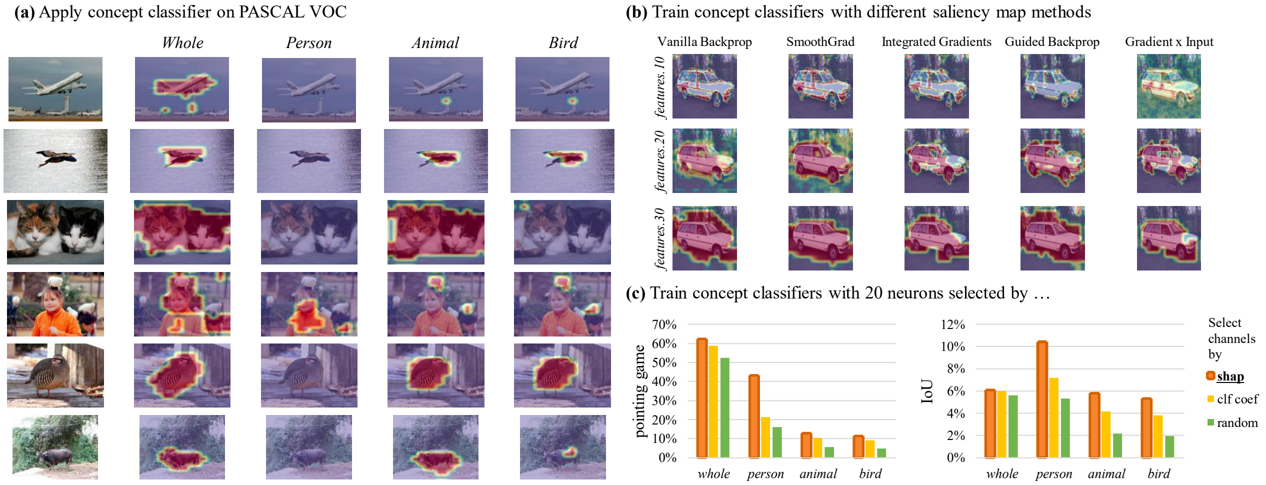

Figure 4. Results of Weakly Supervised Object Localization and ablation study. **(a)** Illustration of applying different concept classifiers on PASCAL VOC 2007, showing that HINT can locate objects of chosen concepts. **(b)** Ablation study showing the results of different saliency methods. **(c)** Ablation study showing Shapley values are good measures of neurons' contributions. The concept classifiers are trained with 20 neurons selected by different approaches. The pointing game (mask intersection over the groundtruth mask) and IoU (mask intersection over union of masks) scores show the accuracy of *whole*, *person*, *animal*, and *bird* concept classifiers on PASCAL VOC 2007.

existing WSOL methods adapt the model architecture and develop training techniques to highlight the entire extent rather than discriminative parts of object [6, 32, 36, 38, 62, 65]. However, can we effectively achieve this goal without model adaptation and retraining? HINT provides an approach to utilizing the implicit concepts learned by the model. As shown in Figure 4 (c), classifiers of higher-level concepts (*e.g. whole*) tend to draw larger masks on objects than classifiers of lower-level concepts (*e.g. bird*). It is because that the responsible regions of *whole* contain all the features of its subcategories. Naturally, the *whole* classifier tends to activate full object regions rather than object parts.

4.4. Ablation Study

We perform an ablation study to show that HINT is general and can be implemented with different saliency methods, and Shapley values are good measures of neurons' contributions to concepts.

Implementation with different saliency methods. We train concept classifiers with five modified saliency methods (see Supplementary Material). Then, we apply the classifiers to the object localization task. Figure 4 (b) shows that the five saliency methods all perform well. This shows that HINT is general, and different saliency methods can be integrated into HINT,

Shapley values. To test the effectiveness of Shapley values, we train concept classifiers using 20 neurons on layer features.30 of VGG19 by different selection approaches, including Shapley values (denoted as shap), absolute values of linear classifier coefficients (denoted as clf_coef), and random selection (denoted as random). We then use the classifiers to perform localization tasks on PASCAL VOC 2007 (see Figure 4 (c)). Two metrics are used: pointing game (mask intersection over the groundtruth mask, usually used by other interpretation methods) [66] and IoU (mask intersection over the union of masks). The results show that "shap" outperforms "clf_coef" and "random" when locating different targets. This suggests that Shapley value is a good measure of neuron contribution as it considers both the individual and collaborative effects of neurons. In contrast, linear classifier coefficients assume that neurons are independent of each other.

4.5. More Applications

We further demonstrate HINT's usefulness and extensibility by saliency method evaluation, adversarial attack explanation, and COVID19 classification model evaluation (Figure 5). Please see Supplementary Material for details.

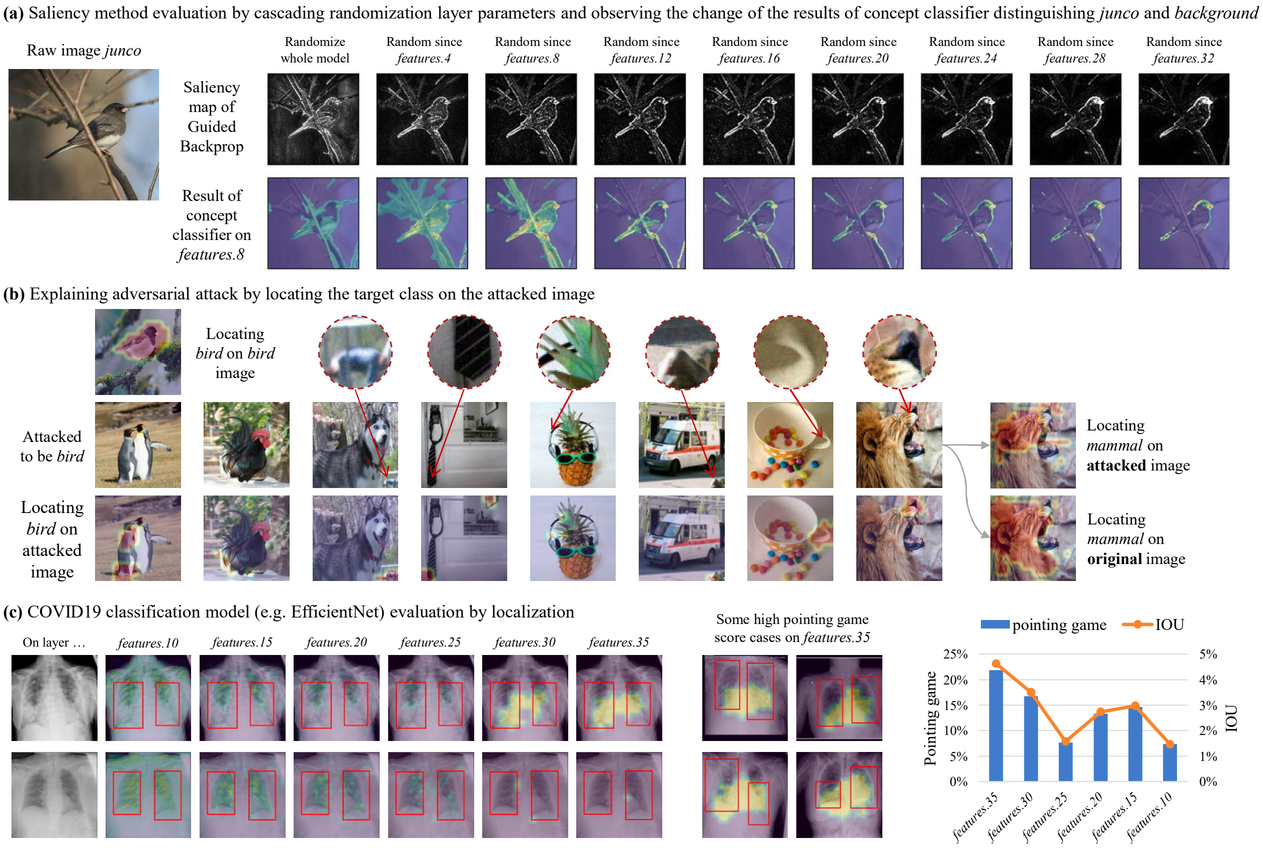
(a) Saliency method evaluation by cascading randomization layer parameters and observing the change of the results of concept classifier distinguishing *junco* and *background*
Raw image *junco*
Randomize whole model
Random since *features.4*
Random since *features.8*
Random since *features.12*
Random since *features.16*
Random since *features.20*
Random since *features.24*
Random since *features.28*
Random since *features.32*
Saliency map of Guided Backprop
Result of concept classifier on *features.8*
(b) Explaining adversarial attack by locating the target class on the attacked image
Locating *bird* on *bird* image
Attacked to be *bird*
Locating *bird* on attacked image
Locating *mammal* on **attacked** image
Locating *mammal* on **original** image
(c) COVID19 classification model (e.g. EfficientNet) evaluation by localization
On layer ...
features.10
features.15
features.20
features.25
features.30
features.35
Some high pointing game score cases on *features.35*
pointing game
IOU
Pointing game
25%
20%
15%
10%
5%
0%
IOU
5%
4%
3%
2%
1%
0%
features.35
features.30
features.25
features.20
features.15
features.10

Figure 5. Other applications of HINT. **(a) Saliency method evaluation.** Guided Backpropagation (GB) can pass the sanity test in [1, 28] if we observe the hidden layer results. With less randomized layers, the classifier-identified regions are more concentrated on the key features of the bird – beak and tail, thereby suggesting that GB detects the salient regions. **(b) Explaining adversarial attack.** We attack images of various classes to be *bird* using PGD [35] and apply the *bird* classifier to their feature map. The responsible regions for concept *bird* highlighted in those fake *bird* images may imply that, some kind of adversarial attacks may be caused by attacking the similar shapes of the target class (*e.g.*, for the coffee mug image where most shapes are round, adversarial attack catches the only pointed shape and attacks it to be like *bird*). **(c) COVID19 classification model evaluation.** Applying deep learning to the detection of COVID19 in chest radiographs has the potential to provide quick diagnosis in resource-limited situations. However, the robustness of those models remains unclear [16]. Object localization with HINT can check whether the identified responsible regions overlap with the lesion regions drawn by doctors.

5. Limitations of Interpretations

HINT can systematically and quantitatively identify the responsible neurons to implicit high-level concepts. However, our approach cannot handle concepts that are not included in the concept hierarchy. It is not effective either to identify responsible neurons to concepts lower than the bottom level of the hierarchy which are the classification categories. More explorations are needed if we want to build such neuron-concept associations.

6. Conclusion

We have presented HIerarchical Neuron concepT explainer (HINT), which builds bidirectional associations between neurons and hierarchical concepts in a low-cost and scalable manner. HINT systematically and quantitatively explains whether and how the neurons learn the high-level hierarchical relationships of concepts implicitly. Moreover, it is able to identify collaborative neurons contributing to the same concept but also the multimodal neurons contributing to multiple concepts. Extensive experiments and applications manifest the effectiveness and usefulness of HINT. We open source our development package and hope HINT could inspire more investigations in this direction.

7. Acknowledgments

This work has been supported in part by Hong Kong Research Grant Council - Early Career Scheme (Grant No. 27209621), HKU Startup Fund, and HKU Seed Fund for Basic Research. We also thank Mr. Zhengzhe Liu for his insightful comments and careful editing of this manuscript.

References

[1] Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. Sanity checks for saliency maps. *arXiv preprint arXiv:1810.03292*, 2018. 8

[2] Sarah Adel Bargal, Andrea Zunino, Vitali Petsiuk, Jianming Zhang, Kate Saenko, Vittorio Murino, and Stan Sclaroff. Guided zoom: Questioning network evidence for fine-grained classification. In *British Machine Vision Conference (BMVC)*, 2019. 2

[3] Muhammad Aurangzeb Ahmad, Carly Eckert, and Ankur Teredesai. Interpretable machine learning in healthcare. In *Proceedings of the 2018 ACM international conference on bioinformatics, computational biology, and health informatics*, 2018. 1

[4] Anish Athalye, Logan Engstrom, Andrew Ilyas, and Kevin Kwok. Synthesizing robust adversarial examples. In *International conference on machine learning*. PMLR, 2018. 1

[5] Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PloS one*, 2015. 3

[6] Wonho Bae, Junhyug Noh, and Gunhee Kim. Rethinking class activation mapping for weakly supervised object localization. In *European Conference on Computer Vision*. Springer, 2020. 6, 7

[7] David Bau, Bolei Zhou, Aditya Khosla, Aude Oliva, and Antonio Torralba. Network dissection: Quantifying interpretability of deep visual representations. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017. 1, 2

[8] David Bau, Jun-Yan Zhu, Hendrik Strobelt, Agata Lapedriza, Bolei Zhou, and Antonio Torralba. Understanding the role of individual units in a deep neural network. *Proceedings of the National Academy of Sciences*, 2020. 1, 2

[9] David Bau, Jun-Yan Zhu, Hendrik Strobelt, Bolei Zhou, Joshua B Tenenbaum, William T Freeman, and Antonio Torralba. Gan dissection: Visualizing and understanding generative adversarial networks. *arXiv preprint arXiv:1811.10597*, 2018. 1, 2

[10] Tom B Brown, Dandelion Mané, Aurko Roy, Martín Abadi, and Justin Gilmer. Adversarial patch. *arXiv preprint arXiv:1712.09665*, 2017. 1

[11] Shan Carter, Zan Armstrong, Ludwig Schubert, Ian Johnson, and Chris Olah. Exploring neural networks with activation atlases. *Distill.*, 2019. 1, 2

[12] Chaofan Chen, Oscar Li, Daniel Tao, Alina Barnett, Cynthia Rudin, and Jonathan K Su. This looks like that: deep learning for interpretable image recognition. *Advances in neural information processing systems*, 32, 2019. 2

[13] Hugh Chen, Scott Lundberg, and Su-In Lee. Explaining models by propagating shapley values of local components. In *Explainable AI in Healthcare and Medicine*. Springer, 2021. 3

[14] Junsuk Choe and Hyunjung Shim. Attention-based dropout layer for weakly supervised object localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019. 6

[15] Piotr Dabkowski and Yarin Gal. Real time image saliency for black box classifiers. *arXiv preprint arXiv:1705.07857*, 2017. 3

[16] Alex J DeGrave, Joseph D Janizek, and Su-In Lee. Ai for radiographic covid-19 detection selects shortcuts over signal. *Nature Machine Intelligence*, 2021. 8

[17] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 2009. 5

[18] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results. http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html. 6

[19] Ruth Fong, Mandela Patrick, and Andrea Vedaldi. Understanding deep networks via extremal perturbations and smooth masks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019. 3

[20] Ruth Fong and Andrea Vedaldi. Net2vec: Quantifying and explaining how concepts are encoded by filters in deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018. 2

[21] Ruth C Fong and Andrea Vedaldi. Interpretable explanations of black boxes by meaningful perturbation. In *Proceedings of the IEEE international conference on computer vision*, 2017. 3

[22] Amirata Ghorbani, James Wexler, James Zou, and Been Kim. Towards automatic concept-based explanations. *arXiv preprint arXiv:1902.03129*, 2019. 2

[23] Amirata Ghorbani and James Zou. Neuron shapley: Discovering the responsible neurons. *arXiv preprint arXiv:2002.09815*, 2020. 1, 2

[24] Mara Graziani, Vincent Andrearczyk, and Henning Müller. Regression concept vectors for bidirectional explanations in histopathology. In *Understanding and Interpreting Machine Learning in Medical Image Computing Applications*. Springer, 2018. 2

[25] Jindong Gu and Volker Tresp. Semantics for global and local interpretation of deep neural networks. *arXiv preprint arXiv:1910.09085*, 2019. 2

[26] Jindong Gu, Yinchong Yang, and Volker Tresp. Understanding individual decisions of cnns via contrastive backpropagation. In *Asian Conference on Computer Vision*. Springer, 2018. 3

[27] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016. 5

[28] Ashkan Khakzar, Sabrina Musatian, Jonas Buchberger, Icxel Valeriano Quiroz, Nikolaus Pinger, Soroosh Baselizadeh, Seong Tae Kim, and Nassir Navab. Towards semantic interpretation of thoracic disease and covid-19 diagnosis models. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2021. 8

[29] Been Kim, Rajiv Khanna, and Oluwasanmi O Koyejo. Examples are not enough, learn to criticize! criticism for interpretability. *Advances in neural information processing systems*, 2016. 1

[30] Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, et al. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In *International conference on machine learning*. PMLR, 2018. 2

[31] Zachary C Lipton. The mythos of model interpretability. int. conf. In *Machine Learning: Workshop on Human Interpretability in Machine Learning*, 2016. 1

[32] Weizeng Lu, Xi Jia, Weicheng Xie, Linlin Shen, Yicong Zhou, and Jinming Duan. Geometry constrained weakly supervised object localization. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVI 16*. Springer, 2020. 6, 7

[33] Scott M Lundberg, Gabriel G Erion, and Su-In Lee. Consistent individualized feature attribution for tree ensembles. *arXiv preprint arXiv:1802.03888*, 2018. 1

[34] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *Proceedings of the 31st international conference on neural information processing systems*, 2017. 1

[35] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017. 8

[36] Jinjie Mai, Meng Yang, and Wenfeng Luo. Erasing integrated learning: A simple yet effective approach for weakly supervised object localization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020. 6, 7

[37] James L McClelland and Timothy T Rogers. The parallel distributed processing approach to semantic cognition. *Nature reviews neuroscience*, 2003. 1

[38] Meng Meng, Tianzhu Zhang, Qi Tian, Yongdong Zhang, and Feng Wu. Foreground activation maps for weakly supervised object localization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021. 6, 7

[39] George A Miller. Wordnet: a lexical database for english. *Communications of the ACM*, 1995. 1, 5

[40] Christoph Molnar. *Interpretable machine learning*. Lulu. com, 2020. 4

[41] Grégoire Montavon, Sebastian Lapuschkin, Alexander Binder, Wojciech Samek, and Klaus-Robert Müller. Explaining nonlinear classification decisions with deep taylor decomposition. *Pattern Recognition*, 2017. 3

[42] Alexander Mordvintsev, Christopher Olah, and Mike Tyka. Inceptionism: Going deeper into neural networks. *Google AI Blog*, 2015. 2

[43] Jesse Mu and Jacob Andreas. Compositional explanations of neurons. *Advances in Neural Information Processing Systems*, 2020. 2

[44] Chris Olah, Alexander Mordvintsev, and Ludwig Schubert. Feature visualization: How neural networks build up their understanding of images. distill, 2018. 1, 2

[45] Chris Olah, Arvind Satyanarayan, Ian Johnson, Shan Carter, Ludwig Schubert, Katherine Ye, and Alexander Mordvintsev. The building blocks of interpretability. *Distill*, 2018. 1, 2

[46] Vitali Petsiuk, Abir Das, and Kate Saenko. Rise: Randomized input sampling for explanation of black-box models. *arXiv preprint arXiv:1806.07421*, 2018. 3

[47] MR Quillian and Semantic Memory'in. Semantic information processing, ed. m. minsky, 1968. 1

[48] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. " 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*, 2016. 1

[49] Lloyd S Shapley. *17. A value for n-person games*. Princeton University Press, 2016. 2, 4

[50] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning important features through propagating activation differences. In *International Conference on Machine Learning*. PMLR, 2017. 3

[51] Avanti Shrikumar, Peyton Greenside, Anna Shcherbina, and Anshul Kundaje. Not just a black box: Learning important features through propagating activation differences. *arXiv preprint arXiv:1605.01713*, 2016. 1, 3

[52] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013. 1, 3

[53] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 5

[54] Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. Smoothgrad: removing noise by adding noise. *arXiv preprint arXiv:1706.03825*, 2017. 1, 3

[55] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. Striving for simplicity: The all convolutional net. *arXiv preprint arXiv:1412.6806*, 2014. 1, 3

[56] Jiawei Su, Danilo Vasconcellos Vargas, and Kouichi Sakurai. One pixel attack for fooling deep neural networks. *IEEE Transactions on Evolutionary Computation*, 2019. 1

[57] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *International Conference on Machine Learning*. PMLR, 2017. 1, 3

[58] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016. 5

[59] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The Caltech-UCSD Birds-200-2011 Dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011. 6

[60] Elizabeth K Warrington. The selective impairment of semantic memory. *The Quarterly journal of experimental psychology*, 1975. 1

[61] Daniel S Weld and Gagan Bansal. The challenge of crafting intelligible intelligence. *Communications of the ACM*, 2019. 1

[62] Haolan Xue, Chang Liu, Fang Wan, Jianbin Jiao, Xiangyang Ji, and Qixiang Ye. Danet: Divergent activation for weakly supervised object localization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019. 6, 7

[63] Jason Yosinski, Jeff Clune, Anh Nguyen, Thomas Fuchs, and Hod Lipson. Understanding neural networks through deep visualization. *arXiv preprint arXiv:1506.06579*, 2015. 2, 6

[64] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*. Springer, 2014. 1, 2, 3, 6

[65] Chen-Lin Zhang, Yun-Hao Cao, and Jianxin Wu. Rethinking the route towards weakly supervised object localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020. 6, 7

[66] Jianming Zhang, Sarah Adel Bargal, Zhe Lin, Jonathan Brandt, Xiaohui Shen, and Stan Sclaroff. Top-down neural attention by excitation backprop. *International Journal of Computer Vision*, 2018. 7

[67] Quanshi Zhang, Ruiming Cao, Feng Shi, Ying Nian Wu, and Song-Chun Zhu. Interpreting cnn knowledge via an explanatory graph. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018. 1

[68] Quanshi Zhang, Yu Yang, Haotian Ma, and Ying Nian Wu. Interpreting cnns via decision trees. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019. 1, 3

[69] Xiaolin Zhang, Yunchao Wei, Jiashi Feng, Yi Yang, and Thomas S Huang. Adversarial complementary learning for weakly supervised object localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018. 6

[70] Xiaolin Zhang, Yunchao Wei, Guoliang Kang, Yi Yang, and Thomas Huang. Self-produced guidance for weakly-supervised object localization. In *Proceedings of the European conference on computer vision (ECCV)*, 2018. 6

[71] Bolei Zhou, David Bau, Aude Oliva, and Antonio Torralba. Interpreting deep visual representations via network dissection. *IEEE transactions on pattern analysis and machine intelligence*, 2018. 1, 2

[72] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Object detectors emerge in deep scene cnns. *arXiv preprint arXiv:1412.6856*, 2014. 1, 2

[73] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016. 6