HUMAN-CENTERED CONCEPT EXPLANATIONS FOR NEURAL NETWORKS

A PREPRINT

Chih-Kuan Yeh[1], Been Kim[2], and Pradeep Ravikumar[1]

[1]Machine Learning Department, Carnegie Mellon University
[2]Google Brain

ABSTRACT

Understanding complex machine learning models such as deep neural networks with explanations is crucial in various applications. Many explanations stem from the model perspective, and may not necessarily effectively communicate why the model is making its predictions at the right level of abstraction. For example, providing importance weights to individual pixels in an image can only express which parts of that particular image are important to the model, but humans may prefer an explanation which explains the prediction by concept-based thinking. In this work, we review the emerging area of concept based explanations. We start by introducing concept explanations including the class of Concept Activation Vectors (CAV) which characterize concepts using vectors in appropriate spaces of neural activations, and discuss different properties of useful concepts, and approaches to measure the usefulness of concept vectors. We then discuss approaches to automatically extract concepts, and approaches to address some of their caveats. Finally, we discuss some case studies that showcase the utility of such concept-based explanations in synthetic settings and real world applications.[1]

1 Introduction

Deep neural networks have been instrumental in many modern Artificial Intelligence (AI) successes, with super-human performance across application areas such as vision recognition, speech recognition, and language understanding [35, 37, 78]. As the performance and complexity of deep neural networks improve, understanding how they operate has also become increasingly difficult. This has led to the burgeoning research area of explainable AI (XAI), which provides tools that enable us to better understand an AI model.

The initial set of XAI methods have focused on providing importance weights for either input features [66, 61, 75, 83], or for training samples [46, 40, 85]. While useful, these explanations may not necessarily effectively communicate why the model is making its predictions *at the right level of abstraction*. For example, provides importance weights to individual pixels in an image can only express which parts of that particular image are important to the model. However, human reasoning often comprises "concept-based thinking," where we can loosely relate concepts to groupings of "similar" examples [5, 76, 52, 63]. Thus, we may obtain better human-centered explanations via *concept-based explanations*, where we explain AI models using human-centered "concepts".

In this work, we review the emerging area of concept based explanations. We start by reviewing common classes of explanations, as well as so-called self-interpretable models that use latent variables as concepts. We then review the class of Concept Activation Vectors (CAV) which characterize concepts using vectors in appropriate spaces of neural activations. We next discuss properties of useful concepts, and approaches to measure the usefulness of concept vectors. We then discuss approaches to automatically extract concepts, and approaches to address some of their caveats. Finally, we discuss some case studies that showcase the utility of such concept-based explanations.

[1]This is a book chapter, and the definitive, peer reviewed and edited version of this article is published in [Neuro-Symbolic Artificial Intelligence: The State of the Art, volume: 342, 337 - 352, 2022, https://ebooks.iospress.nl/DOI/10.3233/FAIA210362].

2 Related Work

While this work is mainly focused on human-centered concept explanations, we start with a brief overview of other classes of explanations. As we will discuss in the sequel, these alternative explanation approaches can in turn be used as a sub-routine within concept based explanations, for instance to explain the importance of individual concepts to the final model output. In Sec. 2.4, we discuss existing work on approaches that use "semantically meaningful" latent variables. These extracted latent variables are similar in spirit to concepts, but with the contrast that they are connected to bespoke generative models or specially designed neural networks, whereas concept based explanations [43] could in principle be provided for general neural network models.

Certain models are inherently interpretable e.g. small decision trees, but for others e.g. deep neural networks, we need to provide post-hoc explanations, which can in principle be applied to any given (or specific classes of) pre-trained models. Most post-hoc explanations fall under the following three categories (note that the categories are not necessarily disjoint): (a) feature-based explanation methods, that attribute the model output to input features, (b) example-based explanation methods, that attribute the model output to training samples, and (c) counterfactual explanations, which answer "what if" questions posed to the model.

2.1 Feature based Explanations

Feature based explanations attribute the model prediction to individual features. A large class of feature-based explanations are based on feature perturbations, and gauge the importance of features by perturbing them and measuring the prediction difference. In this line of work, [87, 24] use perturbations with grey patch occlusions on CNNs, while [89, 14, 22, 56, 19, 23] improve upon these perturbations via generative models and advanced smoothing designs. Gradient-based explanations can also been seen as a variant of perturbation based explanations via infinitesimal perturbations [7, 70, 73, 66], and which range from explicit gradients, to variants that modify back-propagation (such as ignoring negative gradients) to address some caveats with simple gradients. As shown in [3], many recent explanations such as ϵ-LRP [6], Deep LIFT [69], and Integrated Gradients [75] can also be seen as variants of gradient explanations. To reduce the noise in gradient saliency maps, [44] propose removing distractors from the image. SmoothGrad [71] on the other hand generates artificially noisy images via additive Gaussian noise, and averages the gradient of the sampled images; due to the averaging, the resulting output is much less sensitive than vanilla gradients. [61] consider a local region characterized by local perturbations of the test input, and restricted to this region approximate the behavior of the given complex model by a locally linear interpretable model. This has been further specialized to different domains by [86, 57]. [20, 49] further compute the importance of a feature by taking the marginal contribution of the feature to any given subset of the set of all features, and computing a weighted average over all possible subsets; which has roots in cooperative game theory and revenue division.

Any such feature explanation in turn can be related to concept based explanations by simply treating the set of concepts as features to the model, and obtaining the importance weights of the concept-features. We discuss this further in Sec. 4.3

2.2 Sample based Explanations

Sample based explanations attribute the model prediction to individual training samples. Prototype selection methods [9, 42] explain a model by providing a set of "representative" samples from the training data set. [41] additionally provide criticism alongside the prototypes to explain what are not captured by prototypes. [46] provide tractable approximations to estimating *influential* training samples defined as those training examples that are the most helpful for reducing the model prediction loss. This was further accelerated by [34] by using a k-nearest-neighbors based selection over training samples. [4] use a graph over the training samples to select influential training samples. [40] use the Fisher kernel to select important training examples around the model's decision boundary. [85] use a representer theorem [64] for neural networks to decompose the neural network prediction as a sum of weighted kernel similarities between the test point and the training samples, which they then use to derive influential training samples. [59] propose another decomposition but of the loss of the model at the test point based on the NTK kernel between training points and the test point for different weights along the training trajectory.

Any such sample explanation in turn can be related to concept based explanations by associating clusters of the important training examples with concepts [27].

2.3 Counterfactual and Contrastive Explanations

Counterfactual and contrastive explanations [21, 36, 77, 31, 39, 58, 38] answer the question of what to alter in the current input to change the model outcome. Such a contrastive perspective is very amenable to interactive explanations that enable users to understand the model [39, 58]. Such counterfactual explanations can be seen to be related to adversarial examples [12, 29] as they both try to find small perturbations to the test input that changes the model prediction [79]. [80] add group sparsity regularization to improve the semantic structure of such adversarial perturbations. [62] deems a set of features important if the model prediction does not change a lot when only perturbing features not in the set. In related work, [38] deem a set of features as important if perturbations within the set of features changes the model outcome, while perturbations outside the set do not.

Any such counterfactual and contrastive explanation in turn can be related to concept based explanations by asking the question "what concepts to alter to change the outcome of the model". This is explored by a line of recent works [47, 30, 74, 26], and which we discuss further in Sec. 3.

2.4 Semantically Meaningful Latent Variables and Self-Interpretable Concept Networks

A line of work focuses on deriving semantically meaningful latent variables or features, which can naturally be related to concepts. A popular class of these approaches use dimensionality reduction methods (such as variational encoders [45]) to derive latent features that can be connected to human-relatable concepts [13, 45], and which has shown great success in applications such as speech [17] and language [60, 33]. [48] however, show that meaningful latent features cannot be derived in a completely unsupervised setting, and suggest the necessity of appropriate inductive biases.

Another line of work focuses on designing "self-interpretable" models that themselves have semantically-meaningful latent variables. [15] develop such a self-interpretable model for image classification by relating representative training patches to latent variables, while [68] develop ones for language data. [10] further use "concept networks" in their self-interpretable models. [2] extend such designs to improve their robustness. [16] propose to replace batch normalization in deep neural networks with a concept whitening module. The benefit of such self-interpretable models is that they do not make the common assumption that the concept vectors lie in a linear vector space of some neural network layer activations (which we will expand upon in the sequel). The caveat with these models on the other hand is that they are not applicable as a post-hoc explanation approach for arbitrary models.

3 Concept based Explanations: A Human-centered Approach

Suppose we denote the given model by $f : \mathbb{R}^d \mapsto \mathbb{R}^K$, so that given a test input $\mathbf{x} \in \mathbb{R}^d$, it outputs $f(\mathbf{x}) \in \mathbb{R}^K$ as the K scores for each of K classes. The class-specific output for the k-th class is then given by $f_k(\mathbf{x}) \in \mathbb{R}$. We additionally use $f^{[\ell]}(\mathbf{x}) \in \mathbb{R}^{d^{[\ell]}}$ to denote the l-th layer activation for test point $\mathbf{x}$. A concept can then be loosely defined as a vector that "represents" the collection of activation vectors (for the given model) corresponding to concept relevant inputs. In the next couple of sections, we discuss approaches that formalize this further.

3.1 CAV and TCAV

[43] introduce Concept Activation Vectors (CAV) motivated by a line of research that the linear vector space of neural activations may encode meaningful, insightful information [53, 55]. They define CAV as the normal vector to a hyperplane separating non-concept-example activations from concept-example activations. Such a hyperplane can be obtained by a binary classifier that discriminates between positive and negative examples with respect a given concept. More formally, suppose we are given set P of positive examples and a set N of negative examples with respect a given concept, for a total of n examples. Then the intermediate activations of positive examples is given by $\{f^{[\ell]}(\mathbf{x}) : \mathbf{x} \in P\}$, while those for negative examples is given by $\{f^{[\ell]}(\mathbf{x}) : \mathbf{x} \in N\}$. The concept activation vector c for the given concept can then be obtained as:

$$c = \arg\min_{c'} \min_{b} \sum_{j=1}^{n} L(y_j, f^{[\ell]}(\mathbf{x}_j) \cdot c + b),$$

where $y_j = 1$ if $\mathbf{x}_j \in P$ and $y_j = 0$ if $\mathbf{x}_j \in N$, and L is some loss function for binary classification. The concept vector thus acts as a linear discriminant in the activation vector space between positive and negative examples for a concept.

To quantify how relevant a given concept is to the prediction of the model, [43] propose the TCAV score, which is the fraction of training samples whose model classifier scores increase when the input is infinitesimally moved in the direction of the concept. To state this formally, for a concept c, class label k, and layer ℓ, and using $\mathcal{X}_k$ to denote the set

of training samples with class k, we have:

$$TCAV_{c,k,l} = \left| \left\{ \mathbf{x} \in \mathcal{X}_k : \frac{df_k(\mathbf{x})}{df^{[\ell]}(\mathbf{x})} \cdot c > 0 \right\} \right| / |\mathcal{X}_k|.$$

A concept is considered to be related to a class label k if the TCAV score is significantly different from TCAV scores with random concepts (i.e., where concept examples are random examples).

Given *relative concepts* such finer-grained comparisons (e.g., brown hair vs. black hair), we can then obtain relative CAVs, and relative TCAV scores. These are most useful in determining which among set of concepts is most relevant to the model, in contrast to determining whether a concept is relevant to the model at all [11].

3.2 Interpretable Concept Basis

Similar to CAV, [88] define an *interpretable concept basis* based on the linear discriminant on an intermediate layer $f^{[\ell]}$, obtaining concept vectors $c = \arg\min_{c'} \min_b \sum_{j=1}^n L(y_j, f^{[\ell]}(\mathbf{x}_j) \cdot c + b)$, given binary concept labels $\{y_j\}_{j=1}^n$ for a set of inputs $\{\mathbf{x}_j\}_{j=1}^n$; following the same notation as in the earlier section. Here, they specify the intermediate layer $f^{[\ell]}$ to be the second to last layer just before the logit scores, which are thus given by:

$$f_{\text{logit}}(\mathbf{x}) = W f^{[\ell]}(\mathbf{x}) + b,$$

for a weight matrix W and bias vector b, The prediction logit for class k is then given by:

$$f_{\text{logit};k}(\mathbf{x}) = w_k \cdot f^{[\ell]}(\mathbf{x}) + b_k.$$

The goal of [88] is to then decompose the weight vectors w_k in terms of the concept vectors $\{c_j\}_{j=1}^m$, such that

$$w_k \approx \alpha_{k1} c_1 + \alpha_{k2} c_2 + ... + \alpha_{km} c_m.$$

The weights $\alpha_k := \{\alpha_{kj}\}_{j=1}^m$ are further constrained to be non-negative since negative weights are harder to interpret; and are further constrained to have sparsity $s < m$. To address these desiderata, they propose to solve for weights $\alpha_k := \{\alpha_{kj}\}_{j=1}^m$ via the following optimization problem:

$$\arg \min_{\alpha_k > 0, |\alpha_k|_0 \leq s} \|w_k - \sum_{j=1}^m \alpha_{kj} c_j\|$$

Thus both interpretable concept bases [88] and CAVs [43] follow the same recipe for deriving concept vectors. The main distinction between them is that the former learns concepts vectors from the second-to-last layer, and focus on the use of these concept vectors as an interpretable basis for logit weights; while the latter can be applied to any user-chosen layer, and focuses on the use of these concept vectors to determine their importance to the model prediction via TCAV scores.

3.3 Some Empirical Evidence for Concept-Based Explanations

[43] propose an experiment based on a dataset of images, that have objects, such as cucumbers and cabs, along with a text caption, such as "cucumber" and "cab," on the bottom of the image. By varying the noise in the image and the text caption, they obtain several models that each use only the text caption, or only the object in the image for classification. An example image is shown in Fig 1; here the model may classify this image as a car based either on the object in the image, or the text caption. The approximated ground truth of whether the model depends on the image object or text caption is determined by the performance of the model when only the object or text is provided. For instance, if the model performs well only when the image object is provided, but does not perform well when only text is provided, then the model is determined to depend on the image object.

[43] then conducted a 50-person human study on Amazon Mechanical Turk, and asked users to predict whether the model depends on the text or image object for two models based on saliency map explanations [75, 71]. Of the two models, one model depends on text, and the other model depends on image object. Only 52% of users were able to predict the correct model source when given saliency map explanations, while they were almost always able to predict the correct model source when given TCAV score explanations. This is likely due to TCAV showing the importance of human-defined concepts, while saliency scores show the importance of raw features.

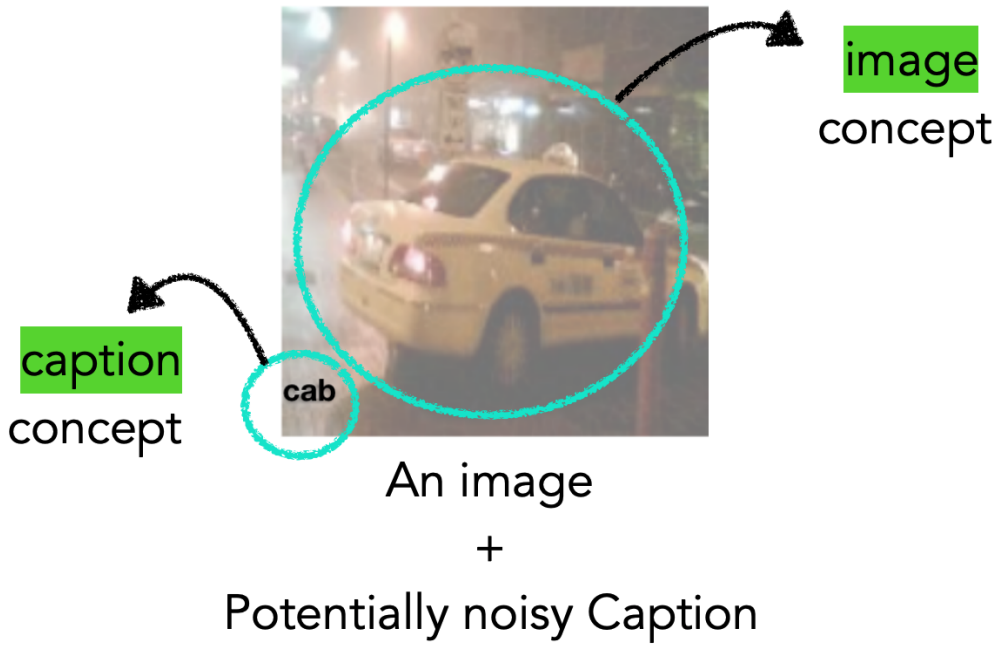

Figure 1: Example image containing both the image of a cab and the text caption of cab.

4 Properties and Evaluation Measures of Concepts

We next discuss some useful properties of concept vectors, as well as measures to evaluate the usefulness of such concept vectors.

4.1 When are Concepts Sufficient to Fully Explain the Model?

A limitation of TCAV scores is that they only answer the question of whether a concept is *related* to the model prediction. However, it may be unclear whether these concepts fully account for the model's prediction. For instance, several concepts may be mildly related to the model prediction, but the main concept critical to the model prediction may not have been discovered. Therefore, it would be useful to determine whether the concepts are also *sufficient* to explain the model prediction.

In this section, we discuss how to measure the sufficiency of concepts as proposed by [84]. The high level assumption here is that the concept scores for a sufficient set of concepts should be a sufficient statistic for the model output. The intuition being that if this does not hold, the model is likely basing its prediction on information *not captured* by the concepts, and thus the concepts are not "sufficient" to explain the model output.

Suppose we are analyzing a set of m concepts, with unit concept vectors $\mathbf{c} := \{\mathbf{c}_j\}_{j=1}^m$ that represent directions in the activation vector space $\text{span}(\text{range}(\phi))) \subseteq \mathbb{R}^d$. Suppose the overall model prediction can be written as $f(\mathbf{x}) = h(\phi(\mathbf{x})))$, where $h(\cdot)$ maps the activations to the model outputs. We also allow for the input $\mathbf{x} = (\mathbf{x}_t)_{t=1}^T$ to correspond to multiple parts (e.g. patches in an image, or some specified sets of input features)

We then define the part concept score for the input part $\mathbf{x}_t$ and concept vectors $\mathbf{c}$ as $v_\mathbf{c}(\mathbf{x}_t) := \langle \phi(\mathbf{x}_t), c_j \rangle_{j=1}^m \in \mathbb{R}^m$, and the *concept score* for the entire input $\mathbf{x}$ as $v_\mathbf{c}(\mathbf{x}) = (v_c)_{t=1}^T \in \mathbb{R}^{T \cdot m}$. These concept scores thus capture similarities of (activations of) the input (parts) to the concept vectors.

We define "sufficient" concepts as those whose concept scores are sufficient statistics for the model output, so that we may evaluate the completeness of concepts by how well one can recover the prediction given the concept scores. Let $g : \mathbb{R}^{T\,m} \to \mathbb{R}^{T\,d}$ denote any mapping from concept scores to the activation space. If concept scores $v_\mathbf{c}(\cdot)$ are sufficient statistics for the model output, then there exists some mapping g_f such that $h(g_f(v_\mathbf{c}(\mathbf{x}))) \approx f(\mathbf{x})$. Following

this intuition, we define the **Completeness Score** $\eta_f(\mathbf{c})$ for a set of concept vectors $\mathbf{c} := \{\mathbf{c}_j\}_{j=1}^m$ as follows:

$$\eta_f(\mathbf{c}_1, ..., \mathbf{c}_m) = \frac{\sup_g \mathbb{P}_{\mathbf{x},y}[y = \arg\max_{y'} h_{y'}(g(v_{\mathbf{c}}(\mathbf{x})))] - a_r}{\mathbb{P}_{\mathbf{x},y}[y = \arg\max_{y'} f_{y'}(\mathbf{x})] - a_r}, \tag{1}$$

where a_r is the accuracy of random prediction to equate the lower bound of completeness score to 0. The population accuracies above can be estimated by the empirical accuracies over validation data. Note that $\sup_g \mathbb{P}_{\mathbf{x},y\sim V}[y = \arg\max_{y'} h_{y'}(g(v_{\mathbf{c}}(\mathbf{x})))]$ is the accuracy of the best possible classifier when only given concept scores $v_{\mathbf{c}}(\mathbf{x})$. In practice, we optimize over a flexible class of classifiers such as DNNs.

Note that the completeness score can also be used to measure how sufficient concepts are for the dataset itself independent of the model, by replacing $\phi(\cdot), h(\cdot)$ with identity functions, and the model prediction $f(\mathbf{x})$ with the label y. We provide an illustrative example below on the usefulness of the completeness score.

Example 4.1. Suppose the inputs $\mathbf{x} \in \mathbb{R}^m$, and the activation function ϕ is simply the identity function. Assume that the features $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_m$ follow independent Bernoulli distributions with bias $p = 0.5$, and the model we attempt to explain is $f(\mathbf{x}) = \mathbf{x}_1$ XOR $\mathbf{x}_2$... XOR $\mathbf{x}_m$. A natural set of m concepts $\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_m$ can be obtained by a one-hot encoding of each feature in $\mathbf{x}$. It can be seen that the full set of concepts $\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_m$ are sufficient for the model prediction, so that $\eta_f(\mathbf{c}_1, \mathbf{c}_2, ..., \mathbf{c}_m) = 1$. However, if we only have information on $\mathbf{c}_1, \mathbf{c}_2, ..., \mathbf{c}_{m-1}$ but not on $\mathbf{c}_m$, then we can only do as well as random guessing in predicting the model output. In this case, $\eta_f(\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_{m-1}) = 0$.

The completeness score allows us assess the 'sufficiency" of the discovered concepts to "explain" the reasoning behind a model's decision. After users manually label some set of concepts and extract concepts of interest in the TCAV score, one can then calculate the concept completeness score. If the completeness score is too low, users could then try to label additional concepts that might be relevant to the model.

4.2 Interventions on Concepts

Given the development of concept vectors as in the previous sections, a natural follow-up question is "how would the model prediction change if a concept were changed in some manner in the input?".

To address this question, [47, 30] propose frameworks to estimate the effect on model output upon intervening on the concepts, with each making different assumptions on the data generative process. [30] assume the generative process follows the graphical model $\mathbf{c} \to \mathbf{x} \to y$, while [47] assume the generative process follows the graphical model $\mathbf{x} \to \mathbf{c} \to y$. Note that the latter explicitly entails that concepts scores are sufficient for the model prediction, which mirrors the sufficiency and completeness viewpoint concepts proposed by [84].

[30] define the Causal Concept Effect (CaCE) as the causal effect of a binary concept $\mathbf{c}$ on the output of the classifier f under the generative process above:

$$\text{CaCE}(\mathbf{c}, f) = \mathbb{E}[f(\mathbf{x}|\text{do}(\mathbf{c} = 1)] - \mathbb{E}[f(\mathbf{x}|\text{do}(\mathbf{c} = 0)].$$

The benefit of the causal concept effect over the TCAV score is that it captures the causal relationship between the concept and model output, while TCAV only captures the associative relationship between the concept and the prediction. Consider the case where the model is solely dependent on the color of the animal (black and white). In the dataset, assume there is an equal number of cat and dogs, but 90% of the cats are white and 90% of the dogs are black. The TCAV model may easily learn that the concept "cat" is related to the model output, since the concept vectors of "cat" and "color" as estimated from this data may be close. However, in spite of this correlation, the causal concept effect may be able to learn that the concept "cat" is not causal to the model prediction. To obtain CaCE values, [30] utilize environments where intervening on concepts is possible, or to approximate these using flexible generative models such as conditional VAE models to generate input conditioning on the concept values.

[47] assume the model prediction score can be computed based solely on the concept scores. They thus train a new model where one of the intermediate layers is exactly the concepts scores for each of the concepts, and use this new model to make predictions based on the concept scores. The benefit of using this retrained model is that it is then possible to intervene on the concept scores, and answer questions such as "If the concept score was different for this input, will the model make a correct prediction?". This also allows user to interact with models and correct certain concept scores for prediction. However, an implicit limitation is that if complete concepts are not provided, it is unclear whether the user would be able to interact with the concept scores as intended.

4.3 Importance Evaluations of Concepts

There are many ways to measure the "importance" of concepts for a set of concepts $\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_m$ and a model $f(\cdot)$. Here, we recap a set of quantitative measurements to evaluate the importance of each concept.

TCAV score [43]: The TCAV score measures the inner product between the concept vector $\mathbf{c}$ and the gradient of the model prediction for class k with respect to activations in an intermediate layer ℓ:

$$TCAV_{c,k,l} = \left| \left\{ \mathbf{x} \in \mathcal{X}_k : \frac{df_k(\mathbf{x})}{df^{[\ell]}(\mathbf{x})} \cdot c > 0 \right\} \right| / |\mathcal{X}_k|.$$

Concept Causal Effect [30]: The concept causal effect of concept $\mathbf{c}$ measures the difference in prediction value of the model f when intervening on value of a binary concept value $\mathbf{c}$:

$$\text{CaCE}(\mathbf{c}, f) = \mathbb{E}[f(\mathbf{x}|\text{do}(\mathbf{c} = 1)] - \mathbb{E}[f(\mathbf{x}|\text{do}(\mathbf{c} = 0)].$$

ConceptSHAP [84]: Given a set of concepts $C_S = \{\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_m\}$, and a completeness score function η, the ConceptSHAP $\mathbf{s}_i$ for concept $\mathbf{c}_i$ is defined as

$$\mathbf{s}_i(\eta) = \sum\nolimits_{S \subseteq C_s \setminus \mathbf{c}_i} \frac{(m - |S| - 1)!|S|!}{m!} [\eta(S \bigcup \{\mathbf{c}_i\}) - \eta(S)],$$

where each concept is treated as a player in a co-operative game. The Shapley value aggregates the contribution of each player based on its performance when co-operating with other players, and uniquely satisfies a natural set of game-theoretic axioms [67, 49].

TCAV^{ICS} [65]: [65] introduce $TCAV^{\text{ICS}}$, which combines TCAV scores with Integrated Gradient (IG) by projecting the IG on the intermediate layer $f^{[\ell]}(\mathbf{x})$ onto the concept direction $\mathbf{c}$. The full formulation can be written as:

$$TCAV^{\text{ICS}}_{\mathbf{c},k,l,b} = \frac{1}{n} \sum_{i=1}^{n} (f^{[\ell]}(\mathbf{x}_i) - b)^T \mathbf{c} \int_{[b, f^{[\ell]}(\mathbf{x}_i)]} \mathbf{c} \cdot \frac{df_k(\mathbf{x}_i)}{df^{[\ell]}(\mathbf{x}_i)} df^{[\ell]}(\cdot).$$

Here, b is a baseline value which is set to be the activation after "removing" concept information, so that we could set $b = f^{[\ell]}(\mathbf{x}) - \lambda \mathbf{c}$, or $b = f^{[\ell]}(\mathbf{x}) - \text{projection}(f^{[\ell]}(\mathbf{x}), \mathbf{c})$.

Each of these concept importance scores can be seen to be related to methods of feature importance, where TCAV is related to the model gradient, CaCE is related to Leave-One-Out explanations (measuring effect on model output when leaving out a feature), ConceptSHAP is related to the Shapley value explanations, and $TCAV^{\text{ICS}}$ is related to the model Integrated Gradient. It would be interesting to see if other approaches for feature importance can be usefully extended to measure the importance of concepts.

5 Unsupervised Discovery of Concepts

One of the main caveats for CAVs is that we need to provide training examples for each of the concept. These typically have to be manually labeled, which however may often not be possible or too costly.

A natural question is whether can one discover concepts in the dataset that are human interpretable in an *unsupervised manner*? We briefly review a few approaches that address this question. [28] discover concepts in images by super-pixel segmentation followed by k-means clustering. [84] discover concepts in image and language data by learning concepts that are representative of clusters of training inputs. [26] find disentangled and diverse concepts (which they term a concept trajectory) by leveraging counterfactuals.

[28] propose the desiderata of "Meaningfulness", "Coherency", "Importance" for concept explanations. To obtain meaningful concepts, they use super-pixel segmentation to mimic the process of humans finding semantically meaningful parts of the image data. To obtain meaningful and coherent parts of the data, they propose to use k-means clustering of the super-pixel segments. Finally, they use TCAV scores to determine if the k-means clusters are related to the model prediction for any specific class. [84] aim to find concepts that are "complete" with respect to the model, while also ensuring the concepts are coherent and meaningful by: (a) limiting each concept to be a contiguous part of the input data (a patch of image or a sub-sentence in language data), and (b) enforcing each concept to be close to its top-k nearest neighbors in the training data. The first constraint increases the meaningfulness of the concept, as contiguous parts of the data are more interpretable. The second constraints also increase human interpretability of the concepts, as users can better understand the concept by observing the top-k nearest neighbor training examples. [26] aim to find k concepts by generating counterfactuals close to the image of interest $\mathbf{x}$. They use a generative model $G_{\mathbf{x}}(\alpha, c)$ they

call DISSECT that generates counterfactuals of any test image $\mathbf{x}$ by only changing the concept c, and where the model prediction on the counterfactual is some predefined value α, so that $f(G_{\mathbf{x}}(\alpha, c)) = \alpha$. They then use this generative model to learn the concepts themselves.

A less ambitious approach might be to simply *propose* potential concepts in an unsupervised manner, and which can then be identified and labeled by human experts.[1] provide a visualization of random directions in the activation space in real-time, and which users can rotate around and label as concepts if they find a direction that is meaningful to them. Another approach is to use statistical test to find interesting directions. While not completely unsupervised, this significantly lowers the labeling cost for concept vectors.

5.1 Undesired Correlation and Dependence between Concepts and Labels

A caveat with the approaches above is that they might learn concepts that are predictive of the class label due to so-called spurious associations and correlations between the concepts and the label, but which are not causally related to the label [50, 51]. A natural reason this might occur is due to what is known as confounding. As an example, consider an MNIST digit classification task of predicting whether a digit is even or odd. Then concept scores for say specific digits of '4' and '5' are predictive of the class label, but are perhaps not the right concepts to be learned. Note that even for a random concept via a random direction in the activation space, the concept scores which are the projection of the input onto the concept direction still encodes predictive information. From a generative process viewpoint, this implies that some confounding variables may exist between input data and concepts, and thus concept scores which are not causally related to the model prediction nonetheless have predictive information.

To account for the confounding variables between input data and concepts, [8] propose the following generative model. Let u denote the confounding variable between the input data $\mathbf{x}$ and the concepts $\mathbf{c}$. Let d denote the unconfounded (unbiased) concepts that are independent of the confounding variable u. The generative model is then given by:

$$d = f_1(y) + \epsilon_1, \tag{2}$$

$$c = d + h(u), \tag{3}$$

$$\mathbf{x} = f_2(u, d) + f_3(y) + \epsilon_2, \tag{4}$$

where ϵ_1, ϵ_2 are independent noise variables, and f_1, f_2, f_3 are deterministic functions. [8] show that debiased concepts from the generative model above can retrieve higher quality concepts in both simulation and real world datasets. Note that even if we use debiased concepts above, they never be fully complete in the sense of [84], as $\mathbf{x}$ may contain more predictive information (due to spurious correlations with the label) when compared to the debiased concepts.

A related idea to the debiasing of concepts is to "whiten" the concepts (so that the covariance of the concepts is the identity matrix). [16] suggest using concept whitening modules within a deep neural network, which they suggest learn concepts that do not rely on the usual assumptions that concept vectors lie in the linear vector space of neural activations [43]. However, [50] show that there is information leakage even when you whiten the concept, since decorrelating concept representations does not remove all statistical dependence. They thus suggest minimizing the mutual information between concept scores to prevent information leakage of soft concept representations.

6 Evaluations of Concept Based Explanations

Given a set of concepts $\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_m$, and a model $f(\cdot)$, how does one evaluate the quality of the set of discovered concepts? We first discuss the setting where the ground truth concepts are unknown, and then provide some case studies where we do know the underlying concepts.

6.1 When Ground Truth Concepts Are Unknown

While most work on concept explanations use specific datasets where ground truth concepts can be retrieved, we next discuss some evaluation measures that can be used even when the ground truth concepts are unknown.

Necessary and Sufficient Concepts [28]: One way to evaluate a set of concepts is to mask out (or only keep) the top k most important concepts in the training data. If the set of concepts is useful for prediction, removing (only keeping) the top-k concepts should result in a large drop (increase) of the model performance.

Completeness Score [84]: The evaluation of the completeness score can also be performed in the absence of knowledge of ground-truth concepts. It can be seen to be similar to the remove and retrain framework above, but where we only partially retrain the part of the model from the intermediate layer activation space to the final prediction. This is because the concept scores of complete concepts are assumed to be a sufficient statistic for the model prediction f.

Synthetic Datasets [81]: [81] introduces the BAM dataset, where objects can be synthesized in different scenes. They then train two models, where one is trained with labels of the objects and one is trained with labels of the scene. They then proposes the model contrast scores (MCS), which measure the difference of concept scores of objects in the two models. The concept score of objects should be larger when the object model is used, and the concept score of scenes should be larger when the scene model is used.

One caveat with the first approach to evaluate concepts is that it involves perturbing the input with respect to the concepts, which may take us out of the data manifold. With model contrast scores, the effect of true concepts could be correctly evaluated, but the concept space and the model space would be limited to those in the BAM dataset. Completeness scores can be used without perturbing images and without synthetic data. However, it does not take into account end-task performance.

6.2 Case Study: Building Post-Hoc Symbolic Explanations for Sequential Decision-Making Problems by User-Defined Concept

Concept-Based explanations have the benefit of speaking the language of users, and can also be used as a base unit to build symbolic explanations with user-defined concepts. We next describe an example of such a partially-symbolic model based on human-defined concepts [74]. In the game Montezuma's Revenge, a popular Atari game, the goal is to score points by gathering jewels and killing enemies along the way. Concepts labeled by humans include "skeleton on left", "left door closed", and "not on ladder". A symbolic approximation of an RL agent can then be learned from the concept scores to the model output with preconditions and cost functions similar to the STRIPS model [25]. The symbolic approximation model chooses an action if the current state satisfies the preconditions of the action, and if the action has the lowest cost function of all viable actions.

Based on this symbolic approximation model, [74] propose to generate contrastive explanations for why the model is not performing a "foil action," which is an alternate plan specified by users. Based on the symbolic action model, the model presents explanations of the form "the foil action A is not chosen because precondition B is not satisfied" or "the foil action A is not chosen because its predicted cost C is higher than the current plan with cost D". As an example, the model could present an explanation that the foil action "go left" is not chosen because precondition "skull not on left" is not satisfied, or that the foil action "attack" is not chosen because the cost of attack is at least 500 when "skull on left", but the cost of the chosen plan is 20. While this proposed algorithm relies on the assumption that it is possible to approximate the applicability of actions and cost functions in terms of high-level concepts and via a symbolic action model, which may be not satisfied for more complex environments, this nevertheless sheds light on how concept explanations can bridge the gap between black-box model and symbolic reasoning as explanations.

6.3 Broader Applications of Concept-Explanations

In addition to extensions in the machine learning community, concept-based explanations have been widely introduced to application areas beyond the machine learning community. In the medical domain, using concept-based explanations [32] show that nuclei texture is a relevant concept in detecting tumor tissue in breast lymph node samples. [18] show that ventricular ejection and filling rates concepts are crucial in cardiac MRI classification. [82] discover that radiomic features describing increased entropy, as well as those describing variations of intensity are useful concepts for the prediction of calcifications. In scientific domains, using concept-based explanations, [72] show that the "Eye" concept is important for the prediction of Category 4 tropical cyclones in a CNN model. [54] show that concepts such as antibiotics and one class of nephrotoxic drugs, non-steroidal anti-inflammatory drugs (NSAIDs), are significant for the prediction of Acute Kidney Injury in time series data.

The adaptation of human-centered explanations in such domains where efficiently communicating with highly skilled experts is crucial showcases the potential of human-centered explanation in real-world problems.

7 Conclusion and Discussion

We have provided an overview of current advances in concept-based explanations to explain complex machine learning models. Concepts can be seen as a way to bridge the gap between the reasoning of humans and machine learning models. Instead of having humans speak the language of machines, in terms of raw input features and training samples, concept-based explanations aim to speak the language of humans, via concepts. Concepts and their scores, as defined Sec. 4.3, can be used as quantitative tool for such translation. We have seen that concept-based methods can be combined with feature-based and sample-based explanations to leverage their strengths, and can also be used as a basis for contrastive explanations. The human-centered nature of concept explanations has also made concept explanation

successful in several real-world applications, especially when highly skilled professionals with established domain concepts (e.g., medicine) are involved.

Despite this exciting progress, many interesting open questions remain. How can we discover unknown confounding concepts? Can we extend concept explanations to models such as tree models, where the assumption that concept vectors are directions in a linear vector space need no longer hold? How can we identify the intrinsic properties of what makes a concept understandable to humans? For instance, in the high dimension activation space in the neural network, why are some vectors/directions understandable to human, while some others are not? Can we teach humans new concepts by identifying training examples that can be discriminated by directions in the activation space? More generally, can we use the richness of concepts to convey knowledge from a super-human-performance model to humans?

References

[1] Kris Sankaran Adrianna Janik. Discovering concepts in learned representations using statistical inference and interactive visualization. *KDD*, 2019.

[2] David Alvarez-Melis and Tommi S Jaakkola. Towards robust interpretability with self-explaining neural networks. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, volume 31, pages 7786–7795, 2018.

[3] Marco Ancona, Enea Ceolini, Cengiz Oztireli, and Markus Gross. A unified view of gradient-based attribution methods for deep neural networks. *International Conference on Learning Representations*, 2018.

[4] Rushil Anirudh, Jayaraman J Thiagarajan, Rahul Sridhar, and Timo Bremer. Influential sample selection: A graph signal processing approach. *arXiv preprint arXiv:1711.05407*, 2017.

[5] Sharon Lee Armstrong, Lila R. Gleitman, and Henry Gleitman. What some concepts might not be. *Cognition*, 13(3):263–308, 1983.

[6] 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*, 10(7):e0130140, 2015.

[7] David Baehrens, Timon Schroeter, Stefan Harmeling, Motoaki Kawanabe, Katja Hansen, and Klaus-Robert MÃžller. How to explain individual classification decisions. *Journal of Machine Learning Research*, 11(Jun):1803–1831, 2010.

[8] Mohammad Taha Bahadori and David Heckerman. Debiasing concept-based explanations with causal analysis. In *International Conference on Learning Representations*, 2020.

[9] Jacob Bien and Robert Tibshirani. Prototype selection for interpretable classification. *The Annals of Applied Statistics*, pages 2403–2424, 2011.

[10] Diane Bouchacourt and Ludovic Denoyer. Educe: Explaining model decisions through unsupervised concepts extraction. *arXiv preprint arXiv:1905.11852*, 2019.

[11] Carrie J. Cai, Emily Reif, Narayan Hegde, Jason D. Hipp, Been Kim, Daniel Smilkov, Martin Wattenberg, Fernanda B. Viégas, Gregory S. Corrado, Martin C. Stumpe, and Michael Terry. Human-centered tools for coping with imperfect algorithms during medical decision-making. *CoRR*, abs/1902.02960, 2019.

[12] Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 39–57. IEEE, 2017.

[13] Tsung-Han Chan, Kui Jia, Shenghua Gao, Jiwen Lu, Zinan Zeng, and Yi Ma. Pcanet: A simple deep learning baseline for image classification? *IEEE transactions on image processing*, 24(12):5017–5032, 2015.

[14] Chun-Hao Chang, Elliot Creager, Anna Goldenberg, and David Duvenaud. Explaining image classifiers by counterfactual generation. In *International Conference on Learning Representations*, 2019.

[15] Chaofan Chen, Oscar Li, Daniel Tao, Alina Barnett, Cynthia Rudin, and Jonathan K Su. This looks like that: deep learning for interpretable image recognition. In *Advances in Neural Information Processing Systems*, pages 8928–8939, 2019.

[16] Zhi Chen, Yijie Bei, and Cynthia Rudin. Concept whitening for interpretable image recognition. *Nature Machine Intelligence*, 2(12):772–782, 2020.

[17] Jan Chorowski, Ron J Weiss, Samy Bengio, and Aäron van den Oord. Unsupervised speech representation learning using wavenet autoencoders. *arXiv preprint arXiv:1901.08810*, 2019.

[18] James R Clough, Ilkay Oksuz, Esther Puyol-Antón, Bram Ruijsink, Andrew P King, and Julia A Schnabel. Global and local interpretability for cardiac mri classification. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 656–664. Springer, 2019.

[19] Piotr Dabkowski and Yarin Gal. Real time image saliency for black box classifiers. In *NeurIPS*, pages 6967–6976, 2017.

[20] Anupam Datta, Shayak Sen, and Yair Zick . Algorithmic transparency via quantitative input influence: Theory and experiments with learning systems. In *Security and Privacy (SP), 2016 IEEE Symposium on*, pages 598–617. IEEE, 2016.

[21] Amit Dhurandhar, Pin-Yu Chen, Ronny Luss, Chun-Chen Tu, Paishun Ting, Karthikeyan Shanmugam, and Payel Das. Explanations based on the missing: Towards contrastive explanations with pertinent negatives. In *Advances in Neural Information Processing Systems*, pages 592–603, 2018.

[22] 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*, pages 2950–2958, 2019.

[23] Ruth Fong and Andrea Vedaldi. Explanations for attributing deep neural network predictions. In *Explainable ai: Interpreting, explaining and visualizing deep learning*, pages 149–167. Springer, 2019.

[24] Ruth C. Fong and Andrea Vedaldi. Interpretable explanations of black boxes by meaningful perturbation. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 3449–3457, 2017.

[25] Hector Geffner and Blai Bonet. A concise introduction to models and methods for automated planning. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 8(1):1–141, 2013.

[26] Asma Ghandeharioun, Been Kim, Chun-Liang Li, Brendan Jou, Brian Eoff, and Rosalind W Picard. Dissect: Disentangled simultaneous explanations via concept traversals. *arXiv preprint arXiv:2105.15164*, 2021.

[27] Amirata Ghorbani, James Wexler, James Zou, and Been Kim. Towards automatic concept-based explanations. In *Advances in Neural Information Processing Systems*, 2019.

[28] Amirata Ghorbani, James Wexler, James Zou, and Been Kim. Towards automatic concept-based explanations. *NeurIPS*, 2019.

[29] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

[30] Yash Goyal, Amir Feder, Uri Shalit, and Been Kim. Explaining classifiers with causal concept effect (cace). *arXiv preprint arXiv:1907.07165*, 2019.

[31] Yash Goyal, Ziyan Wu, Jan Ernst, Dhruv Batra, Devi Parikh, and Stefan Lee. Counterfactual visual explanations. In *International Conference on Machine Learning*, pages 2376–2384, 2019.

[32] 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*, pages 124–132. Springer, 2018.

[33] S. Grover, C. Pulice, G. I. Simari, and V. S. Subrahmanian. Beef: Balanced english explanations of forecasts. *IEEE Transactions on Computational Social Systems*, 6(2):350–364, 2019.

[34] Han Guo, Nazneen Fatema Rajani, Peter Hase, Mohit Bansal, and Caiming Xiong. Fastif: Scalable influence functions for efficient model interpretation and debugging. *arXiv preprint arXiv:2012.15781*, 2020.

[35] 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*, pages 770–778, 2016.

[36] Lisa Anne Hendricks, Ronghang Hu, Trevor Darrell, and Zeynep Akata. Grounding visual explanations. In *ECCV*, 2018.

[37] Geoffrey Hinton, Li Deng, Dong Yu, George E Dahl, Abdel-rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, Tara N Sainath, et al. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal processing magazine*, 29(6):82–97, 2012.

[38] Cheng-Yu Hsieh, Chih-Kuan Yeh, Xuanqing Liu, Pradeep Kumar Ravikumar, Seungyeon Kim, Sanjiv Kumar, and Cho-Jui Hsieh. Evaluations and methods for explanation through robustness analysis. In *International Conference on Learning Representations*, 2020.

[39] S. Joshi, O. Koyejo, Warut D. Vijitbenjaronk, Been Kim, and Joydeep Ghosh. Towards realistic individual recourse and actionable explanations in black-box decision making systems. *ArXiv*, abs/1907.09615, 2019.

[40] Rajiv Khanna, Been Kim, Joydeep Ghosh, and Sanmi Koyejo. Interpreting black box predictions using fisher kernels. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 3382–3390, 2019.

[41] Been Kim, Rajiv Khanna, and Oluwasanmi O Koyejo. Examples are not enough, learn to criticize! criticism for interpretability. In *NeurIPS*, pages 2280–2288, 2016.

[42] Been Kim, Cynthia Rudin, and Julie A Shah. The bayesian case model: A generative approach for case-based reasoning and prototype classification. In *Advances in Neural Information Processing Systems*, pages 1952–1960, 2014.

[43] 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 *ICML*, pages 2673–2682, 2018.

[44] Pieter-Jan Kindermans, Kristof T Schütt, Maximilian Alber, Klaus-Robert Müller, and Sven Dähne. Patternnet and patternlrp–improving the interpretability of neural networks. *International Conference on Learning Representations*, 2018.

[45] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[46] Pang Wei Koh and Percy Liang. Understanding black-box predictions via influence functions. In *ICML*, pages 1885–1894. JMLR. org, 2017.

[47] Pang Wei Koh, Thao Nguyen, Yew Siang Tang, Stephen Mussmann, Emma Pierson, Been Kim, and Percy Liang. Concept bottleneck models. *arXiv preprint arXiv:2007.04612*, 2020.

[48] Francesco Locatello, Stefan Bauer, Mario Lucic, Gunnar Rätsch, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem. Challenging common assumptions in the unsupervised learning of disentangled representations. *arXiv preprint arXiv:1811.12359*, 2018.

[49] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *NeurIPS*, pages 4765–4774, 2017.

[50] Anita Mahinpei, Justin Clark, Isaac Lage, Finale Doshi-Velez, and Weiwei Pan. Promises and pitfalls of black-box concept learning models. *arXiv preprint arXiv:2106.13314*, 2021.

[51] Andrei Margeloiu, Matthew Ashman, Umang Bhatt, Yanzhi Chen, Mateja Jamnik, and Adrian Weller. Do concept bottleneck models learn as intended? *arXiv preprint arXiv:2105.04289*, 2021.

[52] Eric Margolis, Stephen Laurence, et al. *Concepts: core readings*. Mit Press, 1999.

[53] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[54] Diana Mincu, Eric Loreaux, Shaobo Hou, Sebastien Baur, Ivan Protsyuk, Martin Seneviratne, Anne Mottram, Nenad Tomasev, Alan Karthikesalingam, and Jessica Schrouff. Concept-based model explanations for electronic health records. In *Proceedings of the Conference on Health, Inference, and Learning*, pages 36–46, 2021.

[55] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

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

[57] Gregory Plumb, Denali Molitor, and Ameet S Talwalkar. Model agnostic supervised local explanations. In *Advances in Neural Information Processing Systems*, pages 2515–2524, 2018.

[58] Rafael Poyiadzi, Kacper Sokol, Raúl Santos-Rodríguez, T. D. Bie, and Peter A. Flach. Face: Feasible and actionable counterfactual explanations. *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, 2020.

[59] Garima Pruthi, Frederick Liu, Satyen Kale, and Mukund Sundararajan. Estimating training data influence by tracing gradient descent. *Advances in Neural Information Processing Systems*, 33, 2020.

[60] Alec Radford, Rafal Józefowicz, and Ilya Sutskever. Learning to generate reviews and discovering sentiment. *arXiv:1704.01444*, 2017.

[61] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. Why should i trust you?: Explaining the predictions of any classifier. In *KDD*, pages 1135–1144. ACM, 2016.

[62] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. Anchors: High-precision model-agnostic explanations. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[63] Eleanor Rosch. Reclaiming concepts. *Journal of consciousness studies*, 6(11-12):61–77, 1999.

[64] Bernhard Schölkopf, Ralf Herbrich, and Alex J Smola. A generalized representer theorem. In *International conference on computational learning theory*, pages 416–426. Springer, 2001.

[65] Jessica Schrouff, Sebastien Baur, Shaobo Hou, Diana Mincu, Eric Loreaux, Ralph Blanes, James Wexler, Alan Karthikesalingam, and Been Kim. Best of both worlds: local and global explanations with human-understandable concepts. *arXiv preprint arXiv:2106.08641*, 2021.

[66] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantamand, Devi Parikh, and Dhruv Parikh. Grad-cam: Visual explanations from deep networks via gradient-based localization. *International conference on computer vision*, 2017.

[67] Lloyd S. Shapley. *A value for n-person games*, page 31–40. Cambridge University Press, 1988.

[68] Tian Shi, Xuchao Zhang, Ping Wang, and Chandan K Reddy. Corpus-level and concept-based explanations for interpretable document classification. *arXiv preprint arXiv:2004.13003*, 2020.

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

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

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

[72] Conner Sprague, Eric Wendoloski, and Ingrid Guch. Interpretable ai for deep learning- based meteorological applications. In *In American Meteorological Society Annual Meeting. AMS*, 2019.

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

[74] Sarath Sreedharan, Utkarsh Soni, Mudit Verma, Siddharth Srivastava, and Subbarao Kambhampati. Bridging the gap: Providing post-hoc symbolic explanations for sequential decision-making problems with black box simulators. *arXiv preprint arXiv:2002.01080*, 2020.

[75] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 3319–3328. JMLR. org, 2017.

[76] Joshua Brett Tenenbaum. *A Bayesian framework for concept learning*. PhD thesis, Massachusetts Institute of Technology, 1999.

[77] Jasper van der Waa, Marcel Robeer, Jurriaan van Diggelen, Matthieu Brinkhuis, and Mark Neerincx. Contrastive Explanations with Local Foil Trees. In *2018 Workshop on Human Interpretability in Machine Learning (WHI)*, 2018.

[78] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[79] S. Wachter, Brent D. Mittelstadt, and Chris Russell. Counterfactual explanations without opening the black box: Automated decisions and the gdpr. *European Economics: Microeconomics & Industrial Organization eJournal*, 2017.

[80] Kaidi Xu, Sijia Liu, Pu Zhao, Pin-Yu Chen, Huan Zhang, Quanfu Fan, Deniz Erdogmus, Yanzhi Wang, and Xue Lin. Structured adversarial attack: Towards general implementation and better interpretability. *arXiv preprint arXiv:1808.01664*, 2018.

[81] Mengjiao Yang and Been Kim. Benchmarking attribution methods with relative feature importance. *arXiv*, pages arXiv–1907, 2019.

[82] Hugo Yeche, Justin Harrison, and Tess Berthier. Ubs: A dimension-agnostic metric for concept vector interpretability applied to radiomics. In *Interpretability of Machine Intelligence in Medical Image Computing and Multimodal Learning for Clinical Decision Support*, pages 12–20. Springer, 2019.

[83] Chih-Kuan Yeh, Cheng-Yu Hsieh, Arun Sai Suggala, David I. Inouye, and Pradeep Ravikumar. On the (in)fidelity and sensitivity of explanations. In *NeurIPS*, volume abs/1901.09392, 2019.

[84] Chih-Kuan Yeh, Been Kim, Sercan Arik, Chun-Liang Li, Tomas Pfister, and Pradeep Ravikumar. On completeness-aware concept-based explanations in deep neural networks. *Advances in Neural Information Processing Systems*, 33, 2020.

[85] Chih-Kuan Yeh, Joon Kim, Ian En-Hsu Yen, and Pradeep K Ravikumar. Representer point selection for explaining deep neural networks. In *NeurIPS*, pages 9291–9301, 2018.

[86] Rex Ying, Dylan Bourgeois, Jiaxuan You, Marinka Zitnik, and Jure Leskovec. Gnn explainer: A tool for post-hoc explanation of graph neural networks. *arXiv preprint arXiv:1903.03894*, 2019.

[87] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer, 2014.

[88] Bolei Zhou, Yiyou Sun, David Bau, and Antonio Torralba. Interpretable basis decomposition for visual explanation. In *ECCV*, pages 119–134, 2018.

[89] Luisa M. Zintgraf, Taco S. Cohen, Tameem Adel, and Max Welling. Visualizing deep neural network decisions: Prediction difference analysis. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*, 2017.