# Cause and Effect: Concept-based Explanation of Neural Networks

Mohammad Nokhbeh Zaeem[1] and Majid Komeili[2]

***Abstract*— In many scenarios, human decisions are explained based on some high-level concepts. In this work, we take a step in the interpretability of neural networks by examining their internal representation or neuron's activations against concepts. A concept is characterized by a set of samples that have specific features in common. We propose a framework to check the existence of a causal relationship between a concept (or its negation) and task classes. While the previous methods focus on the importance of a concept to a task class, we go further and introduce four measures to quantitatively determine the order of causality. Through experiments, we demonstrate the effectiveness of the proposed method in explaining the relationship between a concept and the predictive behaviour of a neural network.**

## I. Introduction

Applications of Machine Learning (ML) and Artificial Intelligence (AI) as methods to help with automatic decision-making have grown to the extent that it has raised concerns about the trustworthiness of these methods. There have been rules and regulations all around the world that organizations should provide explanations for decisions made by their automated decision-making systems [1]. These concerns often exist whenever the problem at hand is not fully understood, explored or our knowledge of the problem is not complete. Knowing the reasoning of machine learning methods may also help with catching their unwanted behaviours by comparing the reasoning to experts' understanding of problems. On the other hand, explanations can be used to extract the knowledge gained by these black boxes as well. Knowledge extraction can help with a better understanding of the AI view of the problem and the machine learning methods.

Neural networks as one of the most promising forms of AI with high performance on classification problems like ImageNet challenge [2], [3] have been criticized for their black box decision-making process. One of the most important questions asked about neural network's decisions is how a certain *concept* influences the internal representation and eventually the output of the neural network. Here, a concept is a representation of a feature and is defined by a set of samples with that feature, against a random set [4].

Breaking down of the decision, output or *task class* of a given pretrained neural network into high-level humanly meaningful *concepts* presented in the input (post-hoc analysis) have been an active area of research in the past few years. This process is done by inspecting the internal representations or *activations* of the neural network. This approach may be called *concept-based explanation* of neural networks. The training phase of the original network and the explanation phase can be completely separate with different datasets (one for task classes and one for concept classes) and done by different parties. For instance, for predicting the job title of a person from their image, the goal is to determine whether having a clinic as an image's background or a stethoscope around the neck affects the prediction of the job title to be a doctor. These methods do not require concept labels and task labels to be from the same set of samples. For example, the task of predicting the job title of a person is doctor can be represented by a set of physician images. But, the concept clinic may be represented by a set of clinic images.

### A. Nonlinear Concepts

Most concept-based methods often assume that a concept, if present in an activation space, should be linearly separable from non-concept samples [5], [4], [6], [7], [8]. This assumption, however, does not necessarily hold, especially in the earlier layers of a network where the learned features are often not abstract enough to linearly separate concepts [9] or in later layers when they fuse to form higher-level concepts. This hinders these methods' ability in tracking the presence of a concept throughout the network. Another limitation comes from the assumption that the gradient of a section of a network with respect to the input is a good representation of that section [4], [7]. Such a first-order approximation might be misleading. This issue has been extensively discussed for saliency maps –which are also based on gradients approximations– and have been proven misleading [10], [11].

In our method, we check the presence of a concept in a layer's activations by training a concept classifier –a network with the same structure as the task classifier from that layer onward but trained to detect the concept. The accuracy of concept classification gives us a good understanding of the importance and possible influence of the concept on a task class. If in a particular layer, a concept cannot be detected by the concept classifier, it is safe to say that the network cannot recall the concept –i.e. the concept is forgotten (not necessarily universally but to the capacity and power of the given network). Such a conclusion can be made only if the concept classifier shares the same structure as the task classifier since the network structure is the upper limit for extraction power of the network. Moreover, the concept classifier is initialized by the weights of the network under inspection. This initialization will reduce the number of concept samples required for training the concept classifier. This particular

[1] mohammadnokhbehzaeem@carleton.ca
[2]Majid Komeili is with Faculty of Computer Science, Carleton University, 1125 Colonel By Drive, Ottawa, Canada. majid.komeili@carleton.ca

choice of the concept classifier's structure allows us to track the concept information across the original network's layers. This initialization will reduce the number of concept samples required for training the concept classifier. This particular choice of the concept classifier's structure allows us to track the concept information across the original network's layers.

### B. Causality

Another shortcoming of the previous methods is that most methods yield a score that captures the correlation between concepts and output and cannot give any further details about the nature of such a relation [5], [4], [7]. Following the above example about job title classification from images, the correlation between clinic background and being classified as a doctor cannot answer the questions like do all images classified as doctor have clinic background or all images with clinic background are classified as doctors. This problem is sometimes referred to as causality confusion. Note that the goal is not to investigate causal relationships in the training dataset. We aim to investigate the *causal relationship* "learned" by a neural network.

Based on the trained concept classifier and the existing network for task classification, we evaluate whether a concept is necessary, sufficient, or irrelevant for a specific task class. To avoid unnecessary assumptions like linear assumption or first-order approximation, we use a distribution sample set –i.e. a set of samples representing the distribution of data manifold. This set is a representative of the likely inputs of the network. Then we directly measure four relationship scores based on the concept and target predictions for the distribution sample set. The four measures will be extracted in terms of causal expressions, showing whether a concept causes a task class or vice-versa. Unlike the previous works in [5], [7], which are limited to specific network structures like convolutional layers, the proposed method can be applied to a wide range of network structures.

Contributions of this work are as follows. We propose a framework to capture the existence of a given concept in a layer of a neural network without the linear assumption or first-order approximation. We also propose a set of scores to quantify the nature of the relationship between the concept and the network decisions in the form of causal expressions. We show practical applications of our method based on several experiments. Through experiments, we also compare our method with two existing methods, namely TCAV [4] and IBD [7], in determining relationships of concepts and tasks. The results show that our method succeeds in cases previous methods fail.

## II. Related Work

TCAV works based on whether the gradient of the neural network is in the direction of the concept. The direction of the concept is defined as the direction orthogonal to the linear classification decision boundary between concept and non-concept samples. The TCAV score captures the correlation between the network output and the concept and lacks detailed information about the nature of the relationship. Moreover, it assumes that concepts can be represented linearly in the activations space, an assumption that does not necessarily hold [9]. They also represent a section of the network only by its gradient (first-order approximation), which might be misleading. A similar approach has also been explored in methods Net2Vec [12] and Network dissection [5], but they assume that the concepts are aligned with single neurons's activation.

In another work, Interpretable Basis Decomposition for visual explanation (IBD) [7], the authors tried to explain the activations of a neural network by greedily decomposing gradient into some concept directions. They use the resulting decomposition as explanations for the image classification task. One of the drawbacks of such an approach is its linear assumption which comes from the usage of linear decomposition of the gradient in the activation space. Using greedy methods can also potentially result in inaccurate and unstable results. Another limitation of the IBD method [7] is that it can only explain convolutional layers and therefore they had to modify the network that include dense layers.

The linear assumption indicates that a concept in hidden layers corresponds to a vector and the representation of data in each layer is a vector space. Such methods assume that addition, subtraction, scalar product and inner product (as projecting an activation to a concept vector) operations in an activations space are always meaningful. The linear assumption is originated in feature visualization methods. Most feature visualization methods optimize for inputs that maximally activate certain neurons or directions. Early studies on neural network activation space tried to find samples that maximally activate a single neuron to associate a concept to the neuron. In [13] the authors argued that random linear combinations of neurons may also correspond to interpretable meaningful concepts. The general idea of using a linear classifier to check the information of intermediate layers originated in [14]. They proposed to use linear probes – trainable linear classifiers independent of the network – to get an insight into the network representations. In contrast to what was mentioned in [13], in [5], [15] the authors reported that the basis (each neuron) direction activation is more often corresponding to a meaningful concept than just random vectors. Still feature visualization methods, ignore the distribution of the input data which results in inputs that are not consistent with real samples.

Linear interaction of concepts has been even less studied in feature visualization methods. In [15] the authors showed in some cases the addition of two concepts' activations will result in inputs with both concepts present. But they cast doubt on whether this finding is always true. Linear assumption lacks enough evidence to be considered reliable for being the basis of interpretability methods that try to gain the trust of humans and justify neural network decisions.

Some other methods have tried to automatically discover new concepts from neural networks, [16], [8], rather than taking a concept as input. Though these methods can help with cases that no principle exists for rational behaviour of

the network, in many cases, the experts have a good principle about the problem at hand and the principle's concepts are predefined. For example for the prediction of a patient having flu, medical experts know that fever is a symptom, and they want to know exactly what is the relation between the fever and patient being classified as having flu.

Our work relates to CACE [6] in that, both try to address the shortcoming of TCAV [4] by capturing causal expressions. The CACE method [6] measures the influence of concept by the difference of conditional expected values. This requires highly controllable datasets or very accurate generative models that may not be available in practice. Our method relates to works that define and train neural networks with concept-based explanations in mind [9], [17], [18], though our method explains existing pretrained neural networks.

Our work relates to [19] in that both use a specific visual method to examine the influence of different input features on the output of a machine learning model. But our method goes further and inspects the nature of the relationship and quantifies these visualizations. We also consider high-level concepts instead of raw input features.

In the next section, we will propose a framework for a concept-based explanation for neural networks, which simultaneously addresses the linear assumption, first-order approximation and causality confusion issues discussed above.

## III. Framework

### A. Background

Logical expressions are usually expressed as a causality clause in the mathematical notation form of $A \Rightarrow B$. In this notation, phenomenon $A$ is the reason for the phenomenon $B$ and whenever $A$ happens, $B$ will follow.

In fundamental math, concepts are represented by sets. We use the same representation to visualize the relation between concept and task in a neural network. Members of a set are samples that the corresponding feature is present, and not being in the set means the feature is not present.

There can be several possible relations of the concept set $C$ to the task class set $T$. Each of these relations can also be represented as a causality clause.

- Necessary: $T \subseteq C$ ($T \Rightarrow C$).
- Sufficient: $C \subseteq T$ ($C \Rightarrow T$), reverse of necessary.
- Negative Necessary: $C \cap T = \emptyset$ meaning $C$ and $T$ are inconsistent ($C \Rightarrow \neg T$ or $T \Rightarrow \neg C$).
- Negative Sufficient: $C \cup T = M$ meaning either $C$ or $T$ or both should happen ($\neg C \Rightarrow T$ or $\neg T \Rightarrow C$). ($M$ is a set that contains all the elements).

### B. Methodology

In our method, we base the explanations on a certain layer's activations and explain whether and how the concept interacts with a task class based on the activations. We break up the neural network into two sections, the section before the hidden layer ($f(x)$) and the section after the hidden layer ($g_w(z)$). $w$ denoted the trainable parameters of the second section and the whole network can be expressed as $g_w(f(x))$ (Figure 1c).

As shown in Figure 1, we only need two sets of samples for our analysis. (1) Concept set labelled on the concept information only (Figure 1a). (2) Distribution sample set, without any labelling (Figure 1b). Note that access to the original training task data is not required.

For the sake of explaining the proposed method, let us consider a neural network trained on colour-coded handwritten digits. In the training set, a unique colour was assigned to each class, and samples within each class were coloured accordingly. For instance, all 0's in the training set are red, and all 1's are blue, etc. One would expect the network decision to be influenced by the colour as well as the digit itself. We aim to determine, how the concept, i.e. red, influences the decision-making of this neural network.

The first step in our analysis is to check whether the concept is present in the layer. We check if the second section of the network has adequate power and capacity to distinguish the concept set (colour red) in that layer. The number of output neurons are adjusted to match the concept.

For representing the concept, a set of positive and negative concept examples are used, in our case red samples against other colours (Figure 1a). Then the concept classifier with structure of the second section of the neural network is trained ($g(z)$) to distinguish the concept from non-concept activations. As a result, we will have two networks with identical structures but different parameters ($w$ amd $w'$). $g_w(z)$ is the task classifier, whereas $g_{w'}(z)$ is the concept classifier. For learning $g_{w'}$, we initialize the trainable parameters of $g$ as $w$. Note that the parameters of the first section ($f(x)$) do not change while $w'$ is learned.

Other than showing the concept is present and extractable by the network, training another network with the same structure gives us a way to generalize over concept samples. And since now we have generalizable representations of both task classes and concepts, we can proceed with our causal analysis.

Checking if a set is a subset of another, can be easily done by checking the definition. Since we cannot sample every possible instance in our input space, we only check the relationship on a distribution sample set (Figure 1b). Note that this sample set is chosen randomly and it is not specifically selected like prototyping methods [20]. The set $T$ is a subset of set $C$ if every sample in $T$ is also in $C$, which is equivalent to $C$ being a necessary condition for $T$ or $T \Rightarrow C$. Checking the negation of this definition is much easier (just checking that no counter-example exists). For this purpose, a scatterplot is generated by evaluating the task classifier and concept classifier on distribution sample set (each point in the scatterplot is a sample of the set) (Figure 1d). A counter-example, in this case, is a sample in $C$ and not in $T$, equivalently the top left corner of the scatterplot (e.g. a sampled classified as 0 and not red). Note that the points of the scatter plots are only outputs of the task classifier ($g_w(z)$) and concept classifiers ($g_{w'}(z)$) based on the layer and are not necessarily close to true labels. This

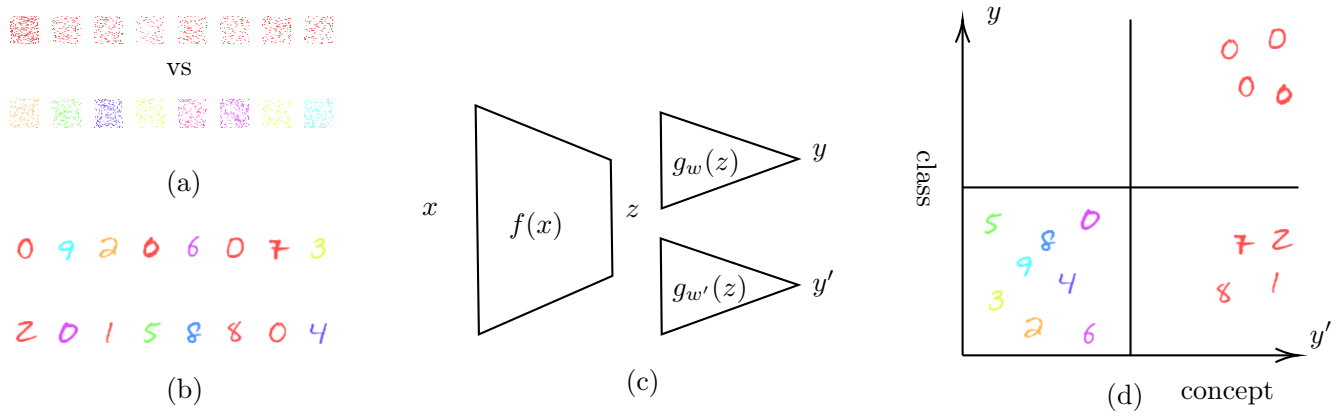


Fig. 1. (a) samples of positive and negative concepts, (b) distribution sample set, (c) representation of task class and concept in network, (d) analysis of the relationship between the task class and concept showing the concept is a necessary condition for the task class.

is a positive point since we want to measure the relationship based on network information and not the true labels.

Two observations support our choice for using the same structure for the detection of concepts. First, if the concept is present and the network is using it, the network has to extract information using its structure so the network structure should be able to detect it. Second, if the concept is not detectable by the existing structure there is no way of it being involved in the network decision. Of course, if the network is not using the concept but it's present in the layer, the evaluation analysis will detect the concept not being involved in task class decision making.

Though concepts like colour can be easily learned by much simpler network structures, more complex concepts like the presence of objects (a stethoscope) might not be as simple to detect. Since this network is pre-trained (on task classification), using it for simpler concepts is not a restriction. Moreover, the structural coherence of the network is kept intact. In other words, the limitations, powers and local behaviour of the network (as initial parameters) are considered in the detection of concept, keeping the convolutional activations as convolution representations (with the spatial information preserved).

### C. Quantifying the Relationships

Since the concept classifier and task class are represented by soft decisions (outputs of the two networks), we propose a method to quantify the absence of counter-examples similar to the ROC curve (see Figure 2).

Consider the fact that the logical expression $C \Rightarrow T$ is equivalent to $C \vee \neg T$. For the expression to be true either $C$ or $\neg T$ has to true. Assuming the threshold $t$ for both expressions and calculating $\neg T$ by $1 - T$, we get the fact that the expression holds for any sample that is not in the $F$ section.

For better handling of imbalanced classes we use the adapted F1 score instead of accuracy:

$$F1 = \frac{2PR}{P+R} \,. \tag{1}$$

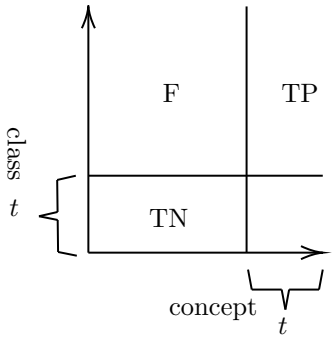


Fig. 2. The process of creating quantification curve for necessary score.

where $P$ and $R$ are adapted precision and recall defined as:

$$P = \frac{TP}{TP+F} \,,\; R = \frac{TN}{TN+F} \,. \tag{2}$$

TP, TN and F denote the number of samples in the corresponding part of Figure 2.

Based on the introduced parameters, for each threshold $t$, an F1 score can be calculated. The measure of the strength of a necessary relationship is then calculated as the area under the F1 versus threshold curve (quantification curve). Intuitively the strongest relationships in this measure, hold with stronger accuracy for smaller thresholds $t$.

For simplicity here, we assumed that the threshold for $C$ and $\neg T$ are equal, but in general, these thresholds can be considered as threshold $t_C$ for $C$ and threshold $t_T$ for $T$. In that case, the quantitative curve will be a 3d surf (the F1 score vs. $t_C$ and $t_T$) and the volume under the curve should be used as the measure of the strength of the relationship.

The four possible relationships mentioned above, respectively, correspond to (1) bottom right, (2) top right, (3) top left, (4) bottom left corners of the Figure 1d being empty. Each of these relationships can logically be converted to an OR ($\vee$) expression and evaluated in the same manner we evaluated the necessary relationship. A simpler way is to logically convert them to a necessary evaluation and quantified with the mentioned process. For instance $C$ being

sufficient for $T$ is equivalent to $T$ being necessary for $C$. The negation of concepts and tasks ($\neg C$) is calculated by just subtracting them from one, i.e. $(1 - C)$, so for negative necessary measurement, we do the same calculation with the negative of concept values.

For any of our experiments, we create four quantification curves based on the scatterplot. We further use the Area Under quantification Curve (AUC) to summarize each curve into a real-valued score between 0 and 1. The area under the curve is a good estimate of how strong the relationship is. For instance, an area under the curve close to one is a very strong relationship while an area of zero means there is no basis for that relationship. In the next section, we demonstrate the proposed methods over several experiments.

## IV. Experiments and Results

In this section, we explore the application of the proposed method in the evaluation of the relationship of neural network task classes and concepts in a controlled setting and a real-world setting. The controlled settings explain neural networks with alexnet structure and the real-world setting explains a pretrained Resnet18. We compare our results with TCAV [4] and IBD [7] methods as they are the most related works to the proposed method.

We construct a dataset by adding a hint caption to two classes of the ImageNet dataset [2], [3], namely class dog and class cat. The hint is added as a white text on the image (by changing the pixels of the image). So part of the sample pixels has some extra information about the class. We consider two scenarios: 1) The caption always reads the same as the image. We call this dataset CaptionDataset1. 2) The caption is always a random word and hence does not include any information about the classification task (dogs vs. cats). We call this dataset CaptionDataset2. The captions have random rotation and scaling associated with them. Figure 3 shows two samples of the CaptionDataset1 images.


Fig. 3. Two samples from CaptionDataset1.

For generating the concept samples (caption concept), we shuffled pixels of images (to wipe out the image information) and then add a caption to the resulting shuffled image. This technique makes sure that the concept is only present in these samples and our representation of the concept is the most accurate.

### A. Analysis of Concept Influence

In this experiment, we show the effectiveness of the proposed method in detecting the causal relationship between a concept and task classes of neural networks. We train a neural network on each of our datasets. The results for CaptionDataset1 and CaptionDataset2 are shown in Figure 4. On the left side, it can be seen that the concept (caption dog) was detected to have a 98% necessary relationship with the class dog. The results for CaptionDataset2 are shown on the right of Figure 4. It can be seen that there is no tangible relationship between the dog class and the caption concept i.e. the AUC of all four measures are small. This confirms that the proposed method detects the causal relationship between the concept and the task classification.

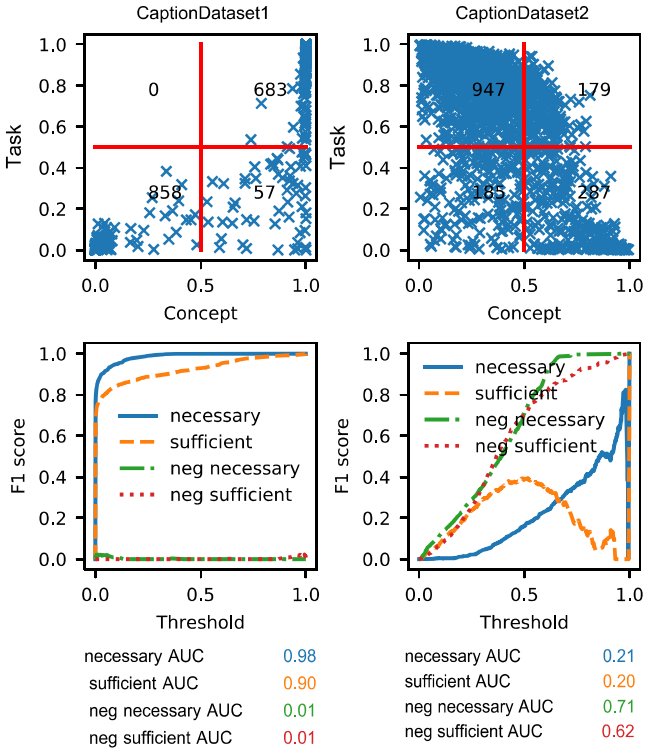


Fig. 4. Comparison of two networks with the same structure, trained on CaptionDataset1 (left side), and CaptionDataset2 (right side). The area of one under the necessary curve shows that the concept is necessary for the task class. The values for each AUC are mentioned at the bottom of the figure. These results are extracted from the last layers of a neural network with an alexnet structure.

Now that we have established that the method can detect the usefulness of the hints, from now on we only use the CaptionDataset1.

### B. Comparison with TCAV Method

In many methods, the directional derivation of a classifier with respect to activations or input is considered a good representation of the model's local behaviour (first-order approximation) [4], [7]. Here we demonstrate that this score is not accurate. We test the score on a neural network trained on our CaptionDataset1. We test the relationship between the dog class and the caption cat. As we know, by design,

these two information are inconsistent in the training data, –i.e. no training data have both. Figure 5 shows that the proposed measures capture the negative relationship between the class dog and the concept caption cat correctly –i.e. the area under the negative necessary (green) curve is very close to one. The third row of Figure 5 shows the distribution of directional derivatives. Though the concept and class are by design inconsistent, directional derivatives are positive on all samples of distribution sample set, showing it is not a reliable explanation. Distribution sample set (points that the evaluations were done) consists of dog and cat images with both dog and cat captions. To show that the linearity of concept classifier does not change this, we repeated the same experiment with linear concept classifier as well (right side of 5).

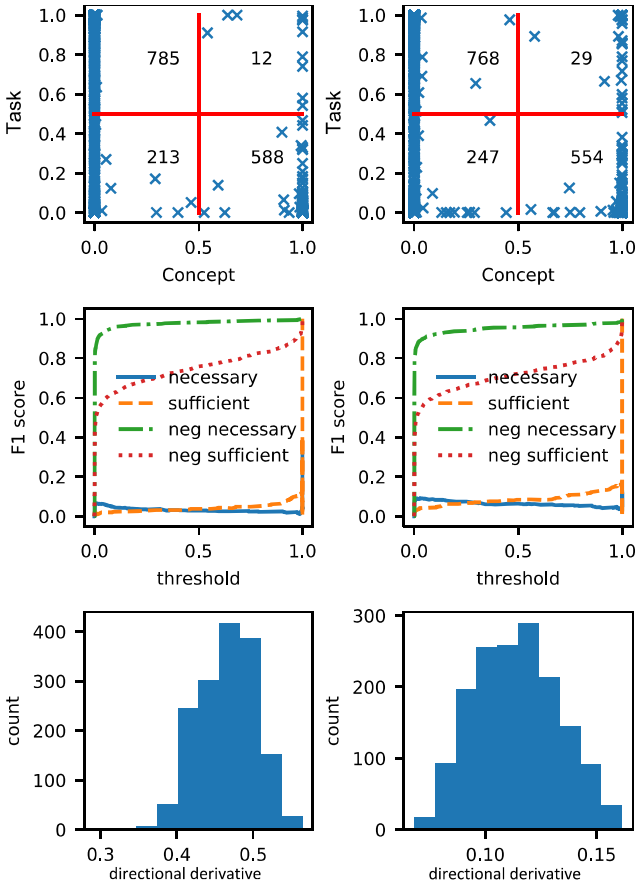


Fig. 5. Results of the caption experiment for linear (right) and nonlinear (left) concept classifiers show that directional derivative can be misleading in both cases. scatterplot (first row). Quantification curve (second row). Distribution of directional derivative (third row).

### C. Comparision with IBD method

Since the IBD method [7] is limited to convolutional networks, in this experiment we change our method to be comparable to IBD. By limiting the proposed method, the concept highlighting as suggested by IBD can be achieved in a similar manner.

We examine the last hidden layer of a Resnet18 trained on the Places365 dataset [21] – a dataset where each class is a place. This network was the benchmark of the IBD method. We use the same set of concept classifiers they trained (with the parameters they provided). We use 10,000 samples from the places365 validation set without their labels as the distribution sample set. Our concepts come from the same dataset IBD method used as their benchmark, Broden [5] – a dataset with segmentation annotations. For a better comparison, we use only use the concepts originally used in the IBD benchmark.

| class | M | Concepts |
|---|---|---|
| topiary garden | P | plant, hedge, tree, brush, flower, bush, sculpture |
| | I | hedge, brush, **tail**, palm, flower, **sheep**, sculpture |
| cross-walk | P | crswlk, road, sidewalk, post, **container**, strtlght, trfc lght |
| | I | crosswalk, minibike, pole, rim, **porch**, cntrl resrv, van |
| indoor market | P | pedestal, sales booth, shop, case, bag, bulletin board, food |
| | I | sales bth, pedestal, food, **fluorescent**, shop, shops, apparel |
| soccer field | P | grass, pitch, grandstand, court, person, post, goal |
| | I | pitch, field, **cage**, **ice rink**, **tennis court**, grass, **tel booth** |
| forest | P | tree, bush, trunk, **cactus**, brush, **fire**, leaves |
| | I | tree, trunk, bush, leaves, **semidesert**, **grid**, **clouds** |
| shoe shop | P | shoe, **bottle**, shelf, box, gym shoe, boot, bag |
| | I | shoe, gym shoe, hndbg, **hat**, catwalk, shop win, **minibike** |
| butte | P | mountain, hill, desert, badlands, rock, valley, land |
| | I | hill, badlands, desert, cliff, **cloud**, mountain, **diffusor** |
| canyon | P | mountain, rock, cliff, hill, badlands, land, desert |
| | I | cliff, mountain, badlands, desert, pond, **bumper**, hill |
| coast | P | sea, sand, land, embankment, rock, **mountain**, water |
| | I | sea, wave, land, **mountain pass**, sand, cliff, **cloud** |
| creek | P | bush, river, rock, land, cliff, tree, earth |
| | I | river, waterfall, land, pond, leaf, **ice**, **fire** |

TABLE I

EXPLANATIONS OF PLACES365 CLASSES USING PROPOSED METHOD AND IBD METHOD. HIGHLIGHTED CONCEPTS WERE CONSIDERED IRRELAVANT TO CLASS BY MAJORITY OF 3 ANNOTATORS.

This experiment is designed to find the most important concepts for classifying each class of the Places365 dataset. For each concept, a concept classifier is trained, and then each task class (a class of Places365) is examined against each concept. The necessary scores of concepts for each task class are sorted and the highest values are reported as the most necessary concepts for the class. The most necessary concepts are then compared against IBD recommended concepts, by decomposition of the decisions into concept space. The top seven are reported for both methods. Then three different annotators were asked to highlight concepts that are not relevant to the class, their majority vote is considered as irrelevant concepts (highlighted in the Table I). The concepts are from left to right in decreasing importance score.

Examining the results of the experiment (by a majority of three annotators), it is apparent that our method assigns more reasonable values of necessary scores to the concepts (compared to what IBD calculates based on its decomposition process). For instance for their benchmark topiary garden, in IBD's top 7 concepts, IBD suggested tail and sheep (among five others) which are irrelevant to the class of topiary garden. On the other hand, our method suggests plant and tree which are quite relevant concepts to the topiary garden class. For soccer field class our method proposes

grass, pitch, grandstand, court, person, post and goal which are all are relevant. But IBD suggests pitch, field, cage, ice rink, tennis court, grass, and telephone booth. Among these concepts, cage, ice rink, tennis court, and telephone booth are irrelevant to the soccer field class (see Table I).

## V. Discussions

The distribution sample set, the set that represents the distribution of likely inputs of the network plays an important role in our analysis as all measure evaluations are based on the samples of this set. Most methods that predict the behaviour of the network need such sample sets, for instance, TCAV [4] need samples from the task class.

The distribution sample set represents likely cases of input and should be a good representation of the inputs that the network will be tested on. The fact that distribution sample set does not need any kind of labelling, enables us to use any set of inputs like a held-out part of data or even inputs recorded from other sources, as long as they are a good representation of likely task classification inputs.

The choice of which layer to inspect is not a straightforward decision. Of course, the inspection of later layers is computationally cheaper (since the training of the concept classifier is cheaper). But there is no guarantee that the concepts are still present in those layers since the network might have traded them with higher levels of abstraction for the task classification. For this reason, we start our analysis from the last layer in the network and work our way back till we reach a layer that the concept is present (classifiable with good accuracy) or reach the first layer (which will guarantee that the concept is too hard to be detected by the network).

## VI. Conclusion

We proposed a framework for verifying the presence of high-level concepts in the activations of the intermediate layers of neural networks. We also determine the type or nature of the causal relationship between a concept and the neural network task classes by quantification of the causal relationship between the task classes and the concept. We showed the effectiveness of the proposed measurements through several comparative experiments, demonstrating improved performance compared with previous methods.

## References


[1] B. Goodman and S. Flaxman, "European union regulations on algorithmic decision-making and a "right to explanation"," *AI Magazine*, vol. 38, no. 3, pp. 50–57, 2017.

[2] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 2009, pp. 248–255.

[3] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, "Imagenet large scale visual recognition challenge," *International journal of computer vision*, vol. 115, no. 3, pp. 211–252, 2015.

[4] B. Kim, M. Wattenberg, J. Gilmer, C. Cai, J. Wexler, F. Viegas, and R. Sayres, "Interpretability beyond feature attribution: Quantitative Testing with Concept Activation Vectors (TCAV)," *35th International Conference on Machine Learning, ICML 2018*, vol. 6, pp. 4186–4195, 2018.

[5] D. Bau, B. Zhou, A. Khosla, A. Oliva, and A. Torralba, "Network dissection: Quantifying interpretability of deep visual representations," in *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, vol. 2017-Janua, 2017, pp. 3319–3327.

[6] Y. Goyal, A. Feder, U. Shalit, and B. Kim, "Explaining classifiers with causal concept effect (cace)," *arXiv preprint arXiv:1907.07165*, 2019.

[7] B. Zhou, Y. Sun, D. Bau, and A. Torralba, "Interpretable basis decomposition for visual explanation," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 11212 LNCS, pp. 122–138, 2018.

[8] C.-K. Yeh, B. Kim, S. Arik, C.-L. Li, T. Pfister, and P. Ravikumar, "On completeness-aware concept-based explanations in deep neural networks," *Advances in Neural Information Processing Systems*, vol. 33, 2020.

[9] P. W. Koh, T. Nguyen, Y. S. Tang, S. Mussmann, E. Pierson, B. Kim, and P. Liang, "Concept bottleneck models," in *International Conference on Machine Learning*. PMLR, 2020, pp. 5338–5348.

[10] J. Adebayo, J. Gilmer, I. Goodfellow, and B. Kim, "Local explanation methods for deep neural networks lack sensitivity to parameter values," *arXiv preprint arXiv:1810.03307*, 2018.

[11] P.-J. Kindermans, S. Hooker, J. Adebayo, M. Alber, K. T. Schütt, S. Dähne, D. Erhan, and B. Kim, "The (un) reliability of saliency methods," *arXiv preprint arXiv:1711.00867*, 2017.

[12] R. Fong and A. 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, pp. 8730–8738.

[13] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," *2nd International Conference on Learning Representations, ICLR 2014 - Conference Track Proceedings*, pp. 1–10, 2014.

[14] G. Alain and Y. Bengio, "Understanding intermediate layers using linear classifier probes," 2016. [Online]. Available: http://arxiv.org/abs/1610.01644

[15] C. Olah, A. Mordvintsev, and L. Schubert, "Feature visualization," *Distill*, vol. 2, no. 11, p. e7, 2017.

[16] A. Ghorbani, J. Wexler, J. Y. Zou, and B. Kim, "Towards automatic concept-based explanations," in *Advances in Neural Information Processing Systems*, 2019, pp. 9277–9286.

[17] M. T. Bahadori and D. E. Heckerman, "Debiasing concept bottleneck models with instrumental variables," *arXiv preprint arXiv:2007.11500*, 2020.

[18] Z. Chen, Y. Bei, and C. Rudin, "Concept whitening for interpretable image recognition," *Nature Machine Intelligence*, vol. 2, no. 12, pp. 772–782, 2020.

[19] J. Wexler, M. Pushkarna, T. Bolukbasi, M. Wattenberg, F. Viégas, and J. Wilson, "The what-if tool: Interactive probing of machine learning models," *IEEE transactions on visualization and computer graphics*, vol. 26, no. 1, pp. 56–65, 2019.

[20] B. Kim, O. Koyejo, R. Khanna *et al.*, "Examples are not enough, learn to criticize! criticism for interpretability." in *NIPS*, 2016, pp. 2280–2288.

[21] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, and A. Torralba, "Places: A 10 million image database for scene recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.