Towards Automating Model Explanations with Certified Robustness Guarantees

Mengdi Huai[1], Jinduo Liu[2], Chenglin Miao[3], Liuyi Yao[4], Aidong Zhang[1]

[1] University of Virginia, [2] Beijing University of Technology, [3] University of Georgia, [4] Alibaba Group
[1]{mh6ck, aidong}@virginia.edu, [2]jinduo@bjut.edu.cn, [3]cmiao@uga.edu, [4]yly287738@alibaba-inc.com

Abstract

Providing model explanations has gained significant popularity recently. In contrast with the traditional feature-level model explanations, concept-based explanations can provide explanations in the form of high-level human concepts. However, existing concept-based explanation methods implicitly follow a two-step procedure that involves human intervention. Specifically, they first need the human to be involved to define (or extract) the high-level concepts, and then manually compute the importance scores of these identified concepts in a post-hoc way. This laborious process requires significant human effort and resource expenditure due to manual work, which hinders their large-scale deployability. In practice, it is challenging to automatically generate the concept-based explanations without human intervention due to the subjectivity of defining the units of concept-based interpretability. In addition, due to its data-driven nature, the interpretability itself is also potentially susceptible to malicious manipulations. Hence, our goal in this paper is to free human from this tedious process, while ensuring that the generated explanations are provably robust to adversarial perturbations. We propose a novel concept-based interpretation method, which can not only automatically provide the prototype-based concept explanations but also provide certified robustness guarantees for the generated prototype-based explanations. We also conduct extensive experiments on real-world datasets to verify the desirable properties of the proposed method.

Introduction

Recently, interpreting and understanding the behaviors of black-box machine learning (ML) models has drawn significant attention (Ghorbani et al. 2019; Kim et al. 2018; Koh et al. 2020; Pedapati et al. 2020; Jeyakumar et al. 2020; Heskes et al. 2020; O'Shaughnessy et al. 2020; Heskes et al. 2020). The most commonly-used explanation method is to explain an ML model's predictions in terms of the input features (e.g., pixels and word-vectors) (Ribeiro, Singh, and Guestrin 2016; Shrikumar, Greenside, and Kundaje 2017; Štrumbelj and Kononenko 2014; Lundberg and Lee 2017; Chen et al. 2018; Koh and Liang 2017). However, these feature-based interpretations suffer from several drawbacks (Ghorbani et al. 2019). For example, (Kim et al. 2018) demonstrates that given identical feature-based explanations, human can confidently find evidence for completely contradicting conclusions. In addition, the feature-based explanation methods are not necessarily the most intuitive explanations for human understanding, especially when using low-level features (e.g., the raw pixels). In contrast, human reasoning often comprises "concept-based thinking" by extracting similarities from numerous samples and grouping them semantically based on their resemblance (Yeh et al. 2019b). As a consequence, recent research has focused on designing concept-based explanation methods to interpret how ML models use high-level human-understandable concepts in arriving at decisions (Kim et al. 2018; Ghorbani et al. 2019; Chen et al. 2019b; Goyal et al. 2019; Wu et al. 2020; Koh et al. 2020; Yeh et al. 2019b; Mincu et al. 2021).

However, an obstacle to the large-scale adoption of these concept-based explanation methods is that they require significant human effort and resource expenditure. The reason is that existing concept-based explanation methods implicitly follow a two-stage procedure with manual intervention. Specifically, they first need human to manually define concepts by using a set of input examples for the ML model under inspection (Ghorbani et al. 2019; Kim et al. 2018; Koh et al. 2020), and then manually compute the importance score of each pre-defined concept in a post-hoc way. For example, to define the concept of "curly", (Ghorbani et al. 2019) needs a human subject to go over all the given images of this concept and extract meaningful segmentations. Then, (Ghorbani et al. 2019) manually computes each extracted concept's importance score via the directional derivative method (Kim et al. 2018). However, identifying human-interpretable concepts and checking for the semantic meaningfulness require a large effort from human experts due to manual annotations and computation. Thus, how to automatically provide concept-based explanations without human intervention still remains a fundamental challenge.

Our goal in this paper is to automatically generate the intrinsic concept-based explanations from the input data without human intervention. To achieve this goal, we propose to inject the concept-based explanations into the learning loop: whenever asking the user to label an incoming sample, the model can simultaneously provide the predicted label for this sample and the corresponding concept-based explanations on interpreting this predicted label. However, the chal-

lenge here is how to define the units of concept-based explanations from the learning network structure considering that they are very subjective. If we directly follow existing concept-based works (Kim et al. 2018; Ghorbani et al. 2019; Chen et al. 2019b; Goyal et al. 2019; Wu et al. 2020; Koh et al. 2020; Yeh et al. 2019b; Mincu et al. 2021), the human has to be involved in this laborious tuning process. The reason is that ML models usually do not comprehend the way humans do and cannot guarantee that the extract concepts are semantically meaningful, which also violates the fidelity of concept-based explanations. So the method proposed in (Ghorbani et al. 2019) still needs human involvement (e.g., removing outliers segments of each concept). In addition, the number of the interpretability units also determines the construction of the self-explanatory network architecture based on the desired properties. Hence, instead of manual tuning, we also need to address how to automatically learn the optimal number of interpretability units.

Furthermore, the concerns regarding the reliability of explanations still exist (Dombrowski et al. 2019). In practice, motivated attackers could generate imperceptible adversarial perturbations to change the interpretability of the input data while preserving the predicted result (Zhang et al. 2020; Ghorbani, Abid, and Zou 2019; Slack et al. 2020; Yeh et al. 2019a). This lack of robustness is problematic in real-world applications where adversarially manipulated explanations could impair safety and trustworthiness. For instance, given a traffic sign classification, a prediction classifying an input as a stop sign with the explanation that the background contains a river is unlikely to be trusted by the users. Although there are some works (Lakkaraju, Arsov, and Bastani 2020; Dombrowski et al. 2019; Mangla, Singh, and Balasubramanian 2020; Soni et al. 2020; Ivankay et al. 2020; Alvarez-Melis and Jaakkola 2018) exploring the robustness of model explanations, they cannot be certified, which means that no provable guarantees can be given to verify their robustness. In practice, these uncertified methods become vulnerable under stronger adversarial attacks. Thus, it is also of great importance to rigorously guarantee robustness of the generated explanations. With such certifiable robustness guarantees for the generated explanations, we need not worry about an adversary with a stronger optimizer, or a more clever algorithm for choosing adversarial perturbations.

In order to tackle the above challenges, in this paper, we design a novel automatic and robust model interpretation method (AutoRMI), a self-explanatory model that can not only automatically provide the concept-based explanations via units that are more understandable to humans than individual features (e.g., pixels) but also provide certified robustness guarantees for the generated explanations. In our proposed method, given that defining the units of concept-based explanations is very subjective, we first propose an interpretability regularization term that guides the model to extract the prototype-based concepts from the training data during the training process, by borrowing the idea from example-based interpretation methods (Koh and Liang 2017; Cai, Jongejan, and Holbrook 2019). However, different from traditional example-based explanations that identify the most responsible training samples to explain a given prediction, each prototype-based concept in our setting is a representative instance that best presents a possibly target set and summarizes the underlying data pattern. Since these prototype-based concepts are extracted in a way that they can represent a set of particular targets in the training data, we can guarantee that these extracted concepts have meaningful and relevant information. Hence, we can release the burden of human from the multifarious manual engagement process. Additionally, to reduce the susceptibility of the generated explanations to adversarial attacks, we also design a novel interval bound propagation based regularization term, which is a bounding technique derived from interval arithmetic (Katz et al. 2017; Ehlers 2017; Sunaga 1958) and is an incomplete method for training verifiably robust models. Specifically, this bounding based regularization term minimizes an upper bound on the maximum difference between any pair of explanation results when the input can be perturbed within a norm-bounded ball, and is computationally efficient since its computational cost is comparable to two forward passes through the network. Extensive experiments on real-world datasets demonstrate the effectiveness of the proposed interpretation method.

Related Work

Concept-based explanations have drawn much attention recently (Kim et al. 2018; Ghorbani et al. 2019; Chen et al. 2019b; Goyal et al. 2019; Wu et al. 2020; Koh et al. 2020; Yeh et al. 2019b; Mincu et al. 2021). Although these concept-based explanation methods are promising, their scalability is limited by the need for "humans-in-the-loop". The methods proposed in (Ghorbani et al. 2019; Kim et al. 2018; Chen et al. 2019b; Mincu et al. 2021) need human to manually define/extract concepts and quantify the importance score of each pre-defined concept in a post-hoc way. (Goyal et al. 2019) directly performs the intervention of adding or removing a concept. (Wu et al. 2020) explains model decisions in terms of the importance of user-defined concepts. (Yeh et al. 2019b; Alvarez-Melis and Jaakkola 2018) need human to be involved to manually pre-define the number of concepts. By assuming the existence of the concept representation, (Koh et al. 2020; Kazhdan et al. 2020; Losch, Fritz, and Schiele 2019) manually define concepts and then use an intermediate set of human-specified concepts to predict the output task label. However, all of the above mentioned works heavily rely on experienced human experts who are expensive and hard to find. Furthermore, the above mentioned works also fail to address the certified robustness guarantees of the generated model explanations.

Recently, there have been a few efforts (Lakkaraju, Arsov, and Bastani 2020; Levine, Singla, and Feizi 2019; Dombrowski et al. 2019; Mangla, Singh, and Balasubramanian 2020; Soni et al. 2020; Ivankay et al. 2020) that have explored the robustness of model explanations. The authors in (Lakkaraju, Arsov, and Bastani 2020) propose a robust post-hoc feature-level explanation framework for constructing a global explanation. (Levine, Singla, and Feizi 2019; Dombrowski et al. 2019; Mangla, Singh, and Balasubramanian 2020; Ivankay et al. 2020) focus on the post-hoc gradient-based interpretation methods (e.g., Saliency Map) that are

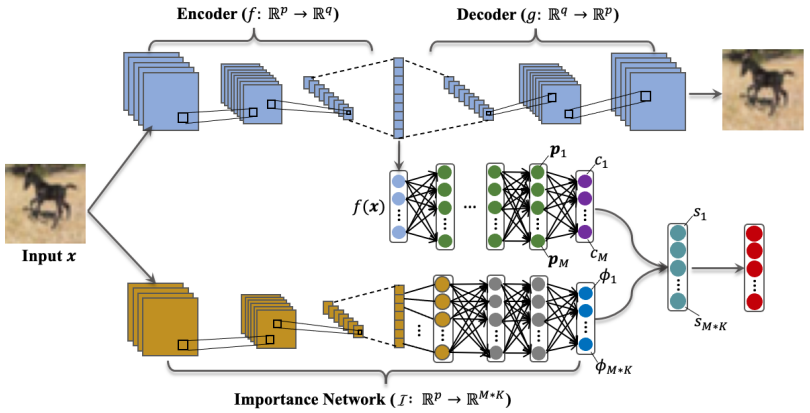


Figure 1: Model structure of the proposed method.

popular methods for deep learning interpretation. (Soni et al. 2020) interprets intermediate layers and defines robustness as the ability of an intermediate layer to be consistent in its recall rate for different random samples. However, their problem settings are significantly different from that of this work, and hence they cannot be directly applied here. In contrast, in this work, we directly guide models to automatically provide the concept-based explanations for each predicted decision, while guaranteeing the robustness of the generated explanations. Although (Alvarez-Melis and Jaakkola 2018) considers the intrinsic interpretation model, it only addresses the stability of the relevance scores and fails to provide robustness guarantees for the obtained interpretable representation. Furthermore, the robustness improvement of all of the aforementioned works cannot be certified – no provable guarantees can be given to verify their robustness. In fact, in practice, these uncertified methods may become vulnerable under stronger adversarial attacks.

Methodology

Note that our goal is to design a novel self-explaining framework, which can not only automatically provide the concept-based explanations without requiring any human intervention but also provide certified robustness guarantees for the generated explanations. However, as aforementioned, defining the units of concept-based explanations is very challenging since they are very subjective. To tackle this challenge, we propose to extract the prototype-based concepts in the training data to guide the model to explain predictions. These learned prototype-based concepts are the representative patterns that describe influential data structures in latent representations. Specifically, we first build an autoencoder component to find the smallest possible representation of data that it can store, and then design a novel interpretation regularizer to extract the prototype-based concepts during training. After that, in order to promote certified robust interpretability, we propose a novel bounding based regularization term. Below, we first give an overview on the model architecture of the proposed method, and then detail the learning objective.

Overview. Formally, we denote the training dataset by $\mathcal{X} = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^N$, where $\boldsymbol{x}_i \in \mathbb{R}^p$ and $y_i \in \{1, \cdots, K\}$. The neural architecture of the proposed AutoRMI is presented in Figure 1. The proposed neural architecture includes an autoencoder network, a concept network, and an importance network. Specifically, the autoencoder network learns a lower-dimension latent representation of the data with an encoder network, $f : \mathbb{R}^p \rightarrow \mathbb{R}^q$. By using the decoder function ($g : \mathbb{R}^q \rightarrow \mathbb{R}^p$), we can project the latent space back to the original dimension. Then, we can pass the learned latent representation (i.e., $f(\boldsymbol{x})$) to the concept network, i.e., $h : \mathbb{R}^q \rightarrow \mathbb{R}^K$. The concept network first uses several fully connected layers over the latent space to learn M prototype-based concepts, i.e., $\{\boldsymbol{p}_m \in \mathbb{R}^q\}_{m=1}^M$. These prototype-based concepts ($\{\boldsymbol{p}_m \in \mathbb{R}^q\}_{m=1}^M$) can provide insight into the representative patterns across the training data that are utilized by the model for predictions. By using the decoder g, we can decode the learned prototype-based concepts to examine what the model has learned. After that, for each $\boldsymbol{p}_m$, the similarity layer computes its distance from the learned latent representation (i.e., $f(\boldsymbol{x})$) as $c_m = ||f(\boldsymbol{x}) - \boldsymbol{p}_m||_2^2$. The smaller the distance value is, the more similar $f(\boldsymbol{x})$ and $\boldsymbol{p}_m$ are. The importance network (i.e., $\mathcal{I} : \mathbb{R}^q \rightarrow \mathbb{R}^{M*K}$) is trained to quantify the importance scores (i.e., $\boldsymbol{\Phi}(\boldsymbol{x}) = [\phi_1(\boldsymbol{x}), \phi_2(\boldsymbol{x}), \cdots, \phi_{M*K}(\boldsymbol{x})]$) of different prototype-based concepts for the predicted result $y(\boldsymbol{x}) = h(f(\boldsymbol{x}))$. Finally, the similarity vector (i.e., $\boldsymbol{c}(\boldsymbol{x})$) and the importance vector (i.e., $\boldsymbol{\Phi}(\boldsymbol{x})$) are aggregated for classification. We use $y(\boldsymbol{x}) = h(f(\boldsymbol{x}))$ to denote the predicted classification result for sample $\boldsymbol{x}$.

The reconstruction error. Note that the autoencoder network here performs data compression and compresses high dimensional data into latent representations via extracting the most prominent features of the original data, and consists of an encoder ($f : \mathbb{R}^p \rightarrow \mathbb{R}^q$) and a decoder ($g : \mathbb{R}^q \rightarrow \mathbb{R}^p$). The input of the encoder is a data sample and its output is the smallest possible latent representation of that sample. The decoder ($g : \mathbb{R}^q \rightarrow \mathbb{R}^p$) takes the latent representation and can project it back to reconstruct the original sample. Here, we use $\tilde{\boldsymbol{x}}_i$ to denote the reconstruction of the original sample $\boldsymbol{x}_i$. The reconstruction loss term for the autoencoder network can be computed as the following sum of the difference between the original input and the consequent reconstruction

$$\mathcal{L}_1(\{\boldsymbol{x}_i\}_{i=1}^N, \{\tilde{\boldsymbol{x}}_i\}_{i=1}^N) = \sum_{i=1}^{N} ||\boldsymbol{x}_i - g(f(\boldsymbol{x}_i))||_2^2, \quad (1)$$

where $\tilde{\boldsymbol{x}}_i$ is the reconstruction of $\boldsymbol{x}_i$. In the above, $g(\cdot)$ and $f(\cdot)$ denote the decoder and encoder, respectively.

The interpretability regularization term. Note that the concept network (i.e., $h : \mathbb{R}^q \rightarrow \mathbb{R}^K$) first learns M concept vectors $\{\boldsymbol{p}_m \in \mathbb{R}^q\}_{m=1}^M$ in the latent space during the training process and then generates a probability distribution over the K classes for each test sample. Due to the subjectivity of defining the units of concept-based explanations, we propose to learn a set of prototype-based concepts (i.e., $\{\boldsymbol{p}_m\}_{m=1}^M$) during model training. These prototype-based concepts are extracted in a way that they can best represent some specific target sets and capture the influential data structures in latent representations, which ensures the semantic meaningfulness of these extracted concepts. With these extracted concepts, we can gain direct insight into representative patterns that are used by the model for classification tasks. For the encoded input $f(\boldsymbol{x}_i)$, the similarity layer computes its squared ℓ_2 distance from each of the prototype-based concepts as $c(\boldsymbol{x}_i) = [c_1 = ||f(\boldsymbol{x}_i) - \boldsymbol{p}_1||_2, \cdots, c_M = ||f(\boldsymbol{x}_i) - \boldsymbol{p}_M||_2]^T$. To enable the model to learn representative patterns from the original input data, we formulate the following interpretability regularization loss term

$$\mathcal{L}_2(\{\boldsymbol{p}_m\}_{m=1}^M, \{\boldsymbol{x}_i\}_{i=1}^N) = \frac{1}{M}\sum_{m=1}^{M} \min_{i\in[1,N]} ||\boldsymbol{p}_m - f(\boldsymbol{x}_i)||_2^2$$
$$+ \frac{1}{N}\sum_{i=1}^{N} \min_{m\in[1,M]} ||f(\boldsymbol{x}_i) - \boldsymbol{p}_m||_2^2 + \frac{2}{M(M-1)}\sum_{m=1}^{M}\sum_{\tilde{m}=m+1}^{M}$$
$$\max(0, d_{min} - ||\boldsymbol{p}_m - \boldsymbol{p}_{\tilde{m}}||_2)^2, \quad (2)$$

where d_{min} is a threshold that classifies whether two prototype-based concepts are close or not. The minimization of the first loss term (i.e., $\frac{1}{M}\sum_{m=1}^{M} \min_{i\in[1,N]} ||\boldsymbol{p}_m - f(\boldsymbol{x}_i)||_2^2$) enforces each prototype-based concept $\boldsymbol{p}_m$ to be as close as possible to at least one of the training examples in the latent space, which will push each prototype-based concept to learn one of the encoded training examples. The minimization of the second loss term is utilized to enforce the encoded training examples in the latent space to be close to one of the concepts, such that the training examples will be clustered around prototypes in the latent space. The third term is a diversity regularization term that exerts a larger penalty on smaller pairwise distances between the prototype-based concepts. By keeping the prototypes distributed in the latent space, it also helps produce a sparser similarity vector.

The misclassification error. Note that for $\boldsymbol{x}_i$, its learned lower-dimension representation in the latent space $f(\boldsymbol{x}_i)$ is passed to the concept network (i.e., $h : \mathbb{R}^q \rightarrow \mathbb{R}^K$) for classification. Specifically, for $\boldsymbol{x}_i$, its learned importance vector (i.e., $\boldsymbol{\Phi}(\boldsymbol{x}_i) = [\phi_1(\boldsymbol{x}_i), \cdots, \phi_{M*K}(\boldsymbol{x}_i)]$) and similarity vector (i.e., $\boldsymbol{c}(\boldsymbol{x})$) are aggregated for classification. Let $h_k(f(\boldsymbol{x}_i))$ denote the probability of $\boldsymbol{x}_i$ belonging to class $k \in [K]$. The cross-entropy loss on the training data (i.e., $\{\boldsymbol{x}_i\}_{i=1}^N$) is given as follows

$$\mathcal{L}_3 = \frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{K} L_y(h_k(f(\boldsymbol{x}_i)), y_i), \quad (3)$$

where L_y is used for penalizing the misclassification.

The certified robust interpretability regularizer. Here, our goal is to provide certified robustness guarantees for the generated explanations (i.e., the importance vector $\boldsymbol{\Phi}(\boldsymbol{x})$). To achieve this goal, we propose to use interval bound propagation to minimize an upper bound on the worst-case loss that any adversarial attack can perturb the generated explanations. Due to space limitations, more discussions on these adversarial attacks are available in the full version of the paper. Note that the input to the importance network (i.e., $\mathcal{I} : \mathbb{R}^q \rightarrow \mathbb{R}^{M*K}$) is denoted $\boldsymbol{x}$ and its output is an importance vector which provides concepts' importance scores. For clarity of presentation, we assume that the importance network is defined by a sequence of transformations h^t for each of its t layers. We use $z^{(t)}$ to denote the output of layer t, where n_t is the number of units in the t-th layer and $z^{(0)}$ stands for the input. Specifically, the network computes

$$z^t = h^{t-1}(z^{(t-1)}), \forall t = 1, \cdots, T, \quad (4)$$

where $z^t \in \mathbb{R}^{M*K}$. Here, we consider the top-k feature robustness, which requires that the set of features with the k highest feature importance scores remains invariant over small-norm adversarial perturbations. In practice, the top-k attack (Ghorbani, Abid, and Zou 2019; Slack et al. 2020; Huai et al. 2020; Sarkar, Sarkar, and Balasubramanian 2020; Stergiou 2021) seeks to perturb the feature importance map by decreasing the relative importance of the k initially most important input features. Let $[D]$ and $S_{\boldsymbol{x},k}$ denote the index set of the input features and the set of features that had the top k highest importance scores for sample $\boldsymbol{x}$, respectively. Let $\bar{S}_{\boldsymbol{x},k} = [D] - S_{\boldsymbol{x},k}$. To produce a certification for the generated explanations (i.e., the importance scores $\boldsymbol{\Phi}(\boldsymbol{x}) = [\phi_1(\boldsymbol{x}), \cdots, \phi_{M*K}(\boldsymbol{x})]$) of sample $\boldsymbol{x}$, we aim to verify the following condition is true

$$\min_{j\in S_{z^{(0)},k}} \underline{\phi}_j(z^{(0)}) - \max_{\bar{j}\in\bar{S}_{z^{(0)},k}} \bar{\phi}_{\bar{j}}(z^{(0)}) \geq 0, \forall z^{(0)} \in \mathbb{B}(\boldsymbol{x}),$$

where $z^{(0)} = \boldsymbol{x}$. Here, $\underline{\phi}_j(\cdot)$ and $\bar{\phi}_j(\cdot)$ denotes the upper and lower bound, respectively. $\mathbb{B}(\boldsymbol{x}) = \{\boldsymbol{x}' | ||\boldsymbol{x} - \boldsymbol{x}'||_\infty \leq \epsilon\}$ is the constraint set over which the adversarial input ranges. Next, we describe how to produce certificates using interval bound propagation as in (Gowal et al. 2018). Specifically, we propose to bound the activation z_t of each layer by an axis-aligned bounding box (i.e., $\underline{z}_t(\epsilon) \leq z_t \leq \bar{z}_t(\epsilon)$[1]) using interval arithmetic. For each coordinate $z_{t,i}$ of z_t, we have

$$\underline{z}_{t,i}(\epsilon) = \min_{\underline{z}_{t-1}(\epsilon)\leq z_{t-1}\leq\bar{z}_{t-1}(\epsilon)} e_i^T h_t(z_{t-1}), \quad (5)$$

$$\bar{z}_{t,i}(\epsilon) = \max_{\underline{z}_{t-1}(\epsilon)\leq z_{t-1}\leq\bar{z}_{t-1}(\epsilon)} e_i^T h_t(z_{t-1}), \quad (6)$$

where $\underline{z}_0(\epsilon) = \boldsymbol{x} - \epsilon\mathbf{1}$, $\bar{z}_0(\epsilon) = \boldsymbol{x} + \epsilon\mathbf{1}$, and e_i is a one-hot vector with 1 in the i-th position. The above optimization

[1]For simplicity, we abuse the notation $\leq$ to mean that all coordinates from the left-hand side need to be smaller than the corresponding coordinates from the right-hand side.

problems can be solved quickly and in closed form for affine layers and monotonic activation functions. Specifically, for the affine layers (e.g., fully connected layers, convolutions) that can be represented in the form $z^t = h^{t-1}(z^{(t-1)}) = W^t z^{t-1} + b^{(t)}$, we can get an outer approximation of the tractable interval range of activations by the next layer z^t using the following formula

$$\bar{z}^{(t)} = W^{(t)}\frac{\bar{z}^{(t-1)} + \underline{z}^{t-1}}{2} + |W^{(t)}|\frac{\bar{z}^{(t-1)} - \underline{z}^{t-1}}{2} + b^{(t)}, \tag{7}$$

$$\underline{z}^{(t)} = W^{(t)}\frac{\bar{z}^{(t-1)} + \underline{z}^{t-1}}{2} - |W^{(t)}|\frac{\bar{z}^{(t-1)} - \underline{z}^{t-1}}{2} + b^{(t)}.$$

Here, $\bar{z}^{(t-1)}$ denotes the upper bound of each interval, $\underline{z}^{(t-1)}$ the lower bound, and $|W^{(t)}|$ the element-wise absolute value. In the similar way, if $h^{(t)}(z^{(t-1)})$ is an element-wise monotonic activation (e.g., a ReLU), then we can calculate the outer approximation of the reachable interval range of the next layer using the following formulas

$$\bar{z}^{(t)} = h^{(t)}(\bar{z}^{(t-1)}), \quad \underline{z}^{(t)} = h^{(t)}(\underline{z}^{(t-1)}). \tag{8}$$

Then, by iteratively applying the above rules, we can propagate intervals through the network and eventually get $\bar{z}^{(T)}$ and $\underline{z}^{(T)}$ (i.e., $\bar{\mathbf{\Phi}}(\boldsymbol{x})$ and $\underline{\mathbf{\Phi}}(\boldsymbol{x})$). A certificate can then be given if we can show that the above verification condition is always true for outputs in the range $\bar{z}^{(T)}$ and $\underline{z}^{(T)}$. Based on this, we propose to minimize the following robustness loss during the training process to provide certified robustness guarantees for the generated explanations

$$\mathcal{L}_4 = \frac{1}{N}\sum_{i=1}^{N}\max(\min_{j\in S_{\boldsymbol{x}_i,k}}\underline{\phi}_j(\boldsymbol{x}_i) - \max_{\bar{j}\in\bar{S}_{\boldsymbol{x}_i,k}}\bar{\phi}_{\bar{j}}(\boldsymbol{x}_i), 0), \tag{9}$$

where $\bar{S}_{\boldsymbol{x}_i,k} = [D] - S_{\boldsymbol{x}_i,k}$.

Full objective. To summarize, the overall loss that we are minimizing is

$$\begin{aligned}\mathcal{L} = &\frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{K}L_y(h_k(f(\boldsymbol{x}_i)), y_i) + \lambda_1\sum_{i=1}^{N}||\boldsymbol{x}_i - g(f(\boldsymbol{x}_i))||_2^2 \\ &+ \frac{\lambda_2}{M}\sum_{m=1}^{M}\min_{i\in[1,N]}||\boldsymbol{p}_m - f(\boldsymbol{x}_i)||_2^2 + \frac{\lambda_3}{N}\sum_{i=1}^{N}\min_{m\in[1,M]}||f(\boldsymbol{x}_i) - \\ &\boldsymbol{p}_m||_2^2 + \frac{2\lambda_4}{M(M-1)}\sum_{m=1}^{M}\sum_{\tilde{m}=m+1}^{M}\max(0, d_{min} - ||\boldsymbol{p}_m - \boldsymbol{p}_{\tilde{m}}||_2)^2 \\ &+ \lambda_5\frac{1}{N}\sum_{i=1}^{N}\max(\min_{j\in S_{\boldsymbol{x}_i,k}}\underline{\phi}_j(\boldsymbol{x}_i) - \max_{\bar{j}\in\bar{S}_{\boldsymbol{x}_i,k}}\bar{\phi}_{\bar{j}}(\boldsymbol{x}_i), 0),\end{aligned} \tag{10}$$

where λ_1, λ_2, λ_3, λ_4, and λ_5 are the trade-off parameters. When we calculate the third and fourth loss terms that take the minimum distance over the entire training dataset, the gradient computation would grow linearly with the size of the training set. However, this would be impractical during training for a large dataset. To solve this challenge, we propose relaxing the minimization to be over only the random minibatch used by the adopted gradient descent algorithm.

Discussion. In Eqn. (10), the value of M denotes the number of the considered concepts and determines the network structure of the model. If we directly follow existing concept-based explanation works (Koh et al. 2020; Kazhdan et al. 2020; Losch, Fritz, and Schiele 2019) to to manually define M, it will require significant human effort and human involvement. Thus, in order to free human from tedious manual finding of a particular set of concepts (i.e., the value of M) in explaining the model's prediction behavior, we propose to follow the firefly neural architecture descent framework proposed in (Wu, Wang, and Liu 2019) to automatically determine the value of M (i.e., the set of satisfactory prototype-based concepts) via the automatic construction of the self-explanatory network architecture based on the desired properties. In addition, we can also follow the above proposed robust interpretability regularizer to provide certified robustness guarantees for the generated similarity results (i.e., $c(\boldsymbol{x}) = [c_1, \cdots, c_M]^T$).

Experiments

In this section, we conduct experiments to verify the effectiveness of the proposed method. Here we adopt three image datasets: the **MNIST** (LeCun et al. 1998), **CIFAR-10** (Krizhevsky, Hinton et al. 2009), and **AT&T**[2] datasets. All the experiments are repeated for 10 times on different random permutations of the training instances and we report the average results. The statistic information of the adopted datasets is given in Table 1. Due to space limitations, the parameter settings, the description of the network architectures and more experimental results will be given in the full version of the paper.

Table 1: The statistic information of the adopted datasets.

	MNIST	CIFAR-10	AT&T
Dimension	$28 \times 28 \times 1$	$32 \times 32 \times 3$	$92 \times 112 \times 1$
Size	70,000	60,000	400
Classes	10	10	40
#Training	55,000	45,000	250
#Validation	5,000	5,000	50
#Testing	10,000	10,000	100

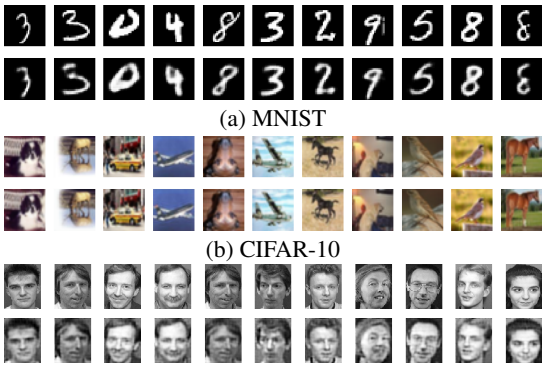
Visualization

Firstly, we evaluate the performance of the trained autoencoder on the adopted datasets. The derived experimental results are reported in Figure 2. Take Figure 2a as an example, where the first line of the images is original images and the second line is the corresponding reconstructed images. From the reported experimental results in this figure, we can see that the reconstructed images are perceptually similar to the original images. We also report the derived reconstruction error and classification accuracy in Table 2. For example, in this table, the testing accuracy of the trained model on the MNIST dataset is 0.9816 and the autoencoder network achieves a reconstruction error of 2.5851, which demonstrates that the proposed AutoRMI can achieve compara-

[2]https://git-disl.github.io/GTDLBench/datasets/att_face_dataset/

	MNIST	CIFAR-10	AT&T
Restruction error	2.5851	3.2088	2.2170
Training Accuracy	0.9887	0.7899	0.8907
Validation Accuracy	0.9924	0.7926	0.9053
Testing Accuracy	0.9816	0.7918	0.8978

Table 2: The reconstruction error and classification accuracy on the adopted datasets.



(a) MNIST

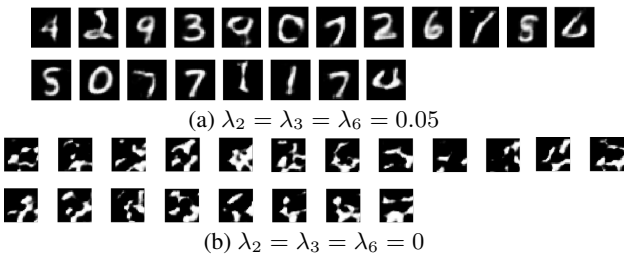
(b) CIFAR-10

(c) AT&T

Figure 2: Reconstructed images on the adopted datasets.

ble accuracy performance to existing classification methods. Importantly, the good performance of the autoencoder component allows us to interpret the learned prototype-based concepts during model training.

Next, we visualize the learned prototype-based concepts. The obtained experimental results on the MNIST dataset are shown in Figure 3. In Figure 3a, we visualize the learned prototype-based concepts (learned in-process during model training) when $\lambda_2 = \lambda_3 = \lambda_6 = 0.05$. Note that these prototype-based concepts are decoded via the decoder. In Figure 3a, we can observe that the prototype-based concepts resemble real-world handwritten digits and give a high-level overview of the original data, due to the designed interpretability regularization term (i.e., $\mathcal{L}_2$ in Eqn. (2)). For comparison, we in Figure 3b also visualize the learned prototype-based concepts when we set $\lambda_2 = \lambda_3 = \lambda_6 = 0$ to remove the interpretability regularization term (i.e., $\mathcal{L}_2$ in Eqn. (2)). From the reported experimental results, we can see that when the interpretability regularizer are removed, the decoded concepts do not look like real-world images, which verifies that the proposed interpretability regularizer can guide the model to learn representative patterns during the training process.

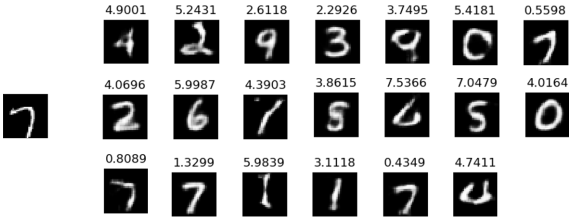


(a) $\lambda_2 = \lambda_3 = \lambda_6 = 0.05$

(b) $\lambda_2 = \lambda_3 = \lambda_6 = 0$

Figure 3: The learned prototype-based concepts on the MNIST dataset where $\lambda_2 = \lambda_3 = \lambda_6 = 0.05$ and $\lambda_2 = \lambda_3 = \lambda_6 = 0$.

Then, we discuss how to use the learned prototype-based concepts to explain each predicted classification result. In Figure 4, we present the visualization results for explaining the predicted classification result for a specific testing image of digit 7, which is shown on the left of Figure 4a. In Figure 4a, we give the distances (computed by the similarity layer) between the encoded representation of this testing image and each of the learned prototype-based concepts, and these distance values are shown above the decoded prototype-based concepts. From Figure 4a, we can see that the three prototype-based concepts that mostly resemble the testing image of digit 7 after decoding have the most shortest distances (i.e., 0.8089, 0.5598, and 0.4349). Importantly, the testing image of digit 7 is more closer to the third "7" concept (in the third line in Figure 4a) than the other two prototype-based concepts. From Figure 4b, we can also observe that compared with other concepts, this most closer concept also has the largest importance score (i.e., 0.999), which verifies that the proposed AutoRMI can capture the subtle differences within the same class. For this specific testing image, its prediction probability of class 7 is 99.98%.



(a)

6.25e-7	6.57e-6	1.04e-7	2.38e-7	2.05e-7	8.94e-8	3.57e-6
2.04e-7	3.75e-7	2.98e-6	1.78e-7	8.90e-7	3.75e-7	5.13e-7
9.23e-6	1.19e-5	1.92e-5	6.55e-7	**0.999**	5.86e-7	-

(b)

Figure 4: Visualization results for the prediction result.

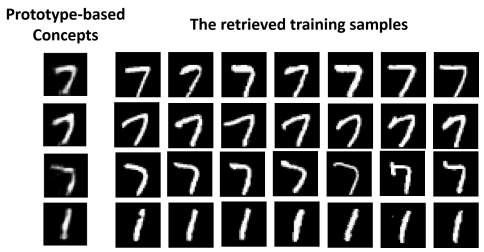


Figure 5: The retrieved training samples having the smallest distances from the learned concepts on the MNIST dataset.

Finally, for each learned prototype-based concept $\boldsymbol{p}_m$, we want to retrieve a subset of training samples that have the shortest distances from this prototype-based concept. Specifically, we aim to represent each learned prototype-based concept $\boldsymbol{p}_m$ by finding a subset of the input training dataset $\boldsymbol{X} = \arg\min_{\hat{\boldsymbol{X}} \subset \mathcal{X}, |\hat{\boldsymbol{X}}|=k} \sum_{\boldsymbol{x} \in \hat{\boldsymbol{X}}} c_m$, where k is pre-defined. Note that the the smaller the value of c_m, the more similar the prototype-based concept $\boldsymbol{p}_m$ and the retrieved training sample, the better the retrieved sample can represent this learned prototype-based concept $\boldsymbol{p}_m$. In this experiment, we set $k = 7$ and select the top-7 closest training examples for each prototype-based concept. The

obtained experimental results on the adopted MNIST dataset are reported in Figure 5. The reported experimental results in this figure show that the learned prototype-based concepts are representative examples. Additionally, from the reported experimental results in this figure, we can also see that the extracted prototype-based concepts and their corresponding retrieved close trained samples are visually similar, which means that they have the same patterns.

Robustness

Evaluation metric. Here, we consider one natural metric for quantifying the similarity between interpretations for two different samples, i.e., the top-k intersection. Specifically, we aim to see how many of the top-k pixels are no longer the top-k pixels after the adversarial perturbation. In many real-world settings, only the most important features are of explanatory interest. In such settings, the attacker can launch the top-k attack (Ghorbani, Abid, and Zou 2019; Slack et al. 2020; Huai et al. 2020; Sarkar, Sarkar, and Balasubramanian 2020; Stergiou 2021; Chen et al. 2019a; Lu et al. 2020; Zhou, Hooker, and Wang 2021; Zhang and Wu 2020; Tursynbek, Petiushko, and Oseledets 2020) to decrease the relative importance of the k initially most important input features. Hence, we propose to compute the size of intersection of the k most important features before and after perturbation.

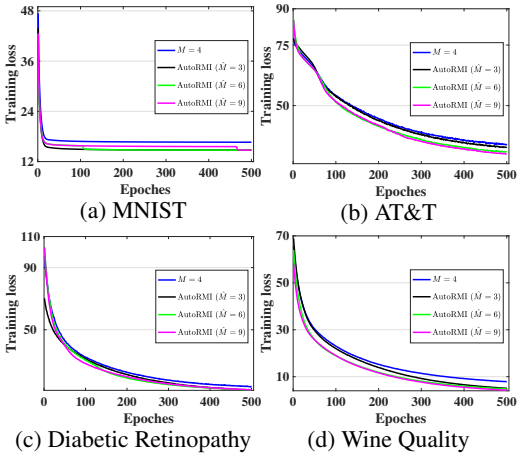
	MNIST		AT&T	
Methods	$\epsilon = 0.5$	$\epsilon = 1.0$	$\epsilon = 0.5$	$\epsilon = 1.0$
AutoRMI	0.89±0.08	0.85±0.10	0.73±0.08	0.67±0.05
Standard	0.79±0.13	0.70±0.06	0.69±0.09	0.62±0.08

Table 3: The change of explanations under different perturbation values.

Performance. To test the empirical robustness of the generated explanations against adversarial perturbations, we here used an ℓ_∞ attack. Here, the value of k is set as 100, and the value of ϵ is varied from 0.5 to 1.0. Since there is no existing certified robustness work, in this experiment, we adopt the standard baseline (denoted as Standard), where we remove the robustness guarantee term (i.e., $\lambda_5 = 0$). In Table 3, we report the obtained experimental results. From the reported experimental results in this table, we can observe that models trained with the proposed certified robust interpretability regularizer in Eqn. (9) perform better than the model obtained with standard training procedure, while the standard model (trained without the certified robust interpretability regularization term) is more vulnerable to adversarial perturbations. Additionally, we can also see that as the severity of adversarial perturbations (the value of ϵ) increases, the networks trained with the proposed method show significant performance improvement over the model trained with standard training process.

Architecture Search

Here, we evaluate the effectiveness of the proposed AutoRMI on the search of the network architectures. We also adopt the Wine Quality and Diabetic Retinopathy datasets (Dua and Graff 2017). We start with a small initial network with $M = 4$ and gradually increase the model size. Note that the value of $\hat{M}$ is the number of new neurons that can be potentially added and hence determines the network structure (e.g., the concept and similarity layers). In Figure 6, we report the training loss of the proposed AutoRMI under different numbers of candidate grown neurons (i.e., different values of $\hat{M}$). Here, the value of $\hat{M}$ is varied from 3 to 9 for the adopted datasets. From this figure, we can see that the objective value gradually converges to 0 when increasing the training epochs, which also verifies that the convergence of AutoRMI can be guaranteed. In addition, we can also observe that the performance improves by even adding three new neurons. Furthermore, the reported experimental results demonstrate that the models trained with the selected network architectures perform better than that trained only with the initial values of M (i.e., $M = 4$ and $\hat{M} = 0$).



(a) MNIST (b) AT&T

(c) Diabetic Retinopathy (d) Wine Quality

Figure 6: The training loss of the proposed method under values of $\hat{M}$ on the adopted datasets.

Conclusions

In this paper, we designed a novel automatic and robust self-explanatory method (AutoRMI) that can not only automatically provide the concept-based explanations without human interventions but also provide certified robustness guarantees for the generated concept-based explanations. Specifically, to free human from the tedious manual defining procedure, we first proposed a novel interpretability regularizer that guides the model to automatically extract the prototype-based concepts from the training data, which provide insights into representative patterns that are utilized by the model for classification. In addition, to promote certified robust interpretability, we also proposed a novel interval bound propagation based regularizer, which minimizes an upper bound on the maximum difference between any pair of explanation results when the input can be adversarially perturbed to provide verifiable robustness guarantees for the generated explanations. We also conducted experiments to demonstrate the effectiveness of the proposed method on real-world datasets and the experimental results show that our method can consistently achieve good performance.

Acknowledgments

This work is supported in part by the US National Science Foundation under grants IIS-1938167, IIS-1955151, IIS-2008208, IIS-2106913, and OAC-1934600. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

References

Alvarez-Melis, D.; and Jaakkola, T. S. 2018. Towards robust interpretability with self-explaining neural networks. *arXiv preprint arXiv:1806.07538*.

Cai, C. J.; Jongejan, J.; and Holbrook, J. 2019. The effects of example-based explanations in a machine learning interface. In *Proceedings of the 24th International Conference on Intelligent User Interfaces*, 258–262.

Chen, J.; Song, L.; Wainwright, M. J.; and Jordan, M. I. 2018. L-Shapley and C-Shapley: Efficient Model Interpretation for Structured Data.

Chen, J.; Wu, X.; Rastogi, V.; Liang, Y.; and Jha, S. 2019a. Robust attribution regularization. *arXiv preprint arXiv:1905.09957*.

Chen, R.; Chen, H.; Ren, J.; Huang, G.; and Zhang, Q. 2019b. Explaining neural networks semantically and quantitatively. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 9187–9196.

Dombrowski, A.-K.; Alber, M.; Anders, C. J.; Ackermann, M.; Müller, K.-R.; and Kessel, P. 2019. Explanations can be manipulated and geometry is to blame. *arXiv preprint arXiv:1906.07983*.

Dua, D.; and Graff, C. 2017. UCI Machine Learning Repository.

Ehlers, R. 2017. Formal verification of piece-wise linear feed-forward neural networks. In *International Symposium on Automated Technology for Verification and Analysis*, 269–286. Springer.

Ghorbani, A.; Abid, A.; and Zou, J. 2019. Interpretation of neural networks is fragile. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, 3681–3688.

Ghorbani, A.; Wexler, J.; Zou, J.; and Kim, B. 2019. Towards automatic concept-based explanations. *Advances in Neural Information Processing Systems*.

Gowal, S.; Dvijotham, K.; Stanforth, R.; Bunel, R.; Qin, C.; Uesato, J.; Arandjelovic, R.; Mann, T.; and Kohli, P. 2018. On the effectiveness of interval bound propagation for training verifiably robust models. *arXiv preprint arXiv:1810.12715*.

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

Heskes, T.; Sijben, E.; Bucur, I. G.; and Claassen, T. 2020. Causal Shapley Values: Exploiting Causal Knowledge to Explain Individual Predictions of Complex Models. *arXiv preprint arXiv:2011.01625*.

Huai, M.; Sun, J.; Cai, R.; Yao, L.; and Zhang, A. 2020. Malicious Attacks against Deep Reinforcement Learning Interpretations. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 472–482.

Ivankay, A.; Girardi, I.; Marchiori, C.; and Frossard, P. 2020. FAR: A General Framework for Attributional Robustness. *arXiv preprint arXiv:2010.07393*.

Jeyakumar, J. V.; Noor, J.; Cheng, Y.-H.; Garcia, L.; and Srivastava, M. 2020. How can i explain this to you? an empirical study of deep neural network explanation methods. *Advances in Neural Information Processing Systems*.

Katz, G.; Barrett, C.; Dill, D. L.; Julian, K.; and Kochenderfer, M. J. 2017. Reluplex: An efficient SMT solver for verifying deep neural networks. In *International Conference on Computer Aided Verification*, 97–117. Springer.

Kazhdan, D.; Dimanov, B.; Jamnik, M.; Liò, P.; and Weller, A. 2020. Now You See Me (CME): Concept-based Model Extraction. *arXiv preprint arXiv:2010.13233*.

Kim, B.; Wattenberg, M.; Gilmer, J.; Cai, C.; Wexler, J.; Viegas, F.; et al. 2018. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In *International conference on machine learning*, 2668–2677. PMLR.

Koh, P. W.; and Liang, P. 2017. Understanding black-box predictions via influence functions. In *International Conference on Machine Learning*. PMLR.

Koh, P. W.; Nguyen, T.; Tang, Y. S.; Mussmann, S.; Pierson, E.; Kim, B.; and Liang, P. 2020. Concept bottleneck models. In *International Conference on Machine Learning*, 5338–5348. PMLR.

Krizhevsky, A.; Hinton, G.; et al. 2009. Learning multiple layers of features from tiny images.

Lakkaraju, H.; Arsov, N.; and Bastani, O. 2020. Robust and stable black box explanations. In *International Conference on Machine Learning*. PMLR.

LeCun, Y.; Bottou, L.; Bengio, Y.; and Haffner, P. 1998. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11): 2278–2324.

Levine, A.; Singla, S.; and Feizi, S. 2019. Certifiably robust interpretation in deep learning. *arXiv preprint arXiv:1905.12105*.

Losch, M.; Fritz, M.; and Schiele, B. 2019. Interpretability beyond classification output: Semantic bottleneck networks. *arXiv preprint arXiv:1907.10882*.

Lu, Y.; Guo, W.; Xing, X.; and Noble, W. S. 2020. Robust Decoy-enhanced Saliency Maps. *arXiv preprint arXiv:2002.00526*.

Lundberg, S. M.; and Lee, S.-I. 2017. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems*, 4765–4774.

Mangla, P.; Singh, V.; and Balasubramanian, V. N. 2020. On Saliency Maps and Adversarial Robustness. *arXiv preprint arXiv:2006.07828*.

Mincu, D.; Loreaux, E.; Hou, S.; Baur, S.; Protsyuk, I.; Seneviratne, M.; Mottram, A.; Tomasev, N.; Karthikesalingam, A.; and Schrouff, J. 2021. Concept-based model explanations for Electronic Health Records. In *Proceedings of the Conference on Health, Inference, and Learning*, 36–46.

O'Shaughnessy, M.; Canal, G.; Connor, M.; Davenport, M.; and Rozell, C. 2020. Generative causal explanations of black-box classifiers. *Advances in Neural Information Processing Systems*.

Pedapati, T.; Balakrishnan, A.; Shanmugam, K.; and Dhurandhar, A. 2020. Learning Global Transparent Models Consistent with Local Contrastive Explanations. *Advances in Neural Information Processing Systems*, 33.

Ribeiro, M. T.; Singh, S.; and Guestrin, C. 2016. Why should i trust you?: Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*.

Sarkar, A.; Sarkar, A.; and Balasubramanian, V. N. 2020. Enhanced Regularizers for Attributional Robustness. *arXiv preprint arXiv:2012.14395*.

Shrikumar, A.; Greenside, P.; and Kundaje, A. 2017. Learning important features through propagating activation differences. In *Proceedings of the 34th International Conference on Machine Learning*.

Slack, D.; Hilgard, S.; Jia, E.; Singh, S.; and Lakkaraju, H. 2020. Fooling lime and shap: Adversarial attacks on post hoc explanation methods. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, 180–186.

Soni, R.; Shah, N.; Seng, C. T.; and Moore, J. D. 2020. Adversarial TCAV–Robust and Effective Interpretation of Intermediate Layers in Neural Networks. *arXiv preprint arXiv:2002.03549*.

Stergiou, A. 2021. The Mind's Eye: Visualizing Class-Agnostic Features of CNNs. *arXiv preprint arXiv:2101.12447*.

Štrumbelj, E.; and Kononenko, I. 2014. Explaining prediction models and individual predictions with feature contributions. *Knowledge and information systems*, 41(3): 647–665.

Sunaga, T. 1958. Theory of an interval algebra and its application to numerical analysis. *RAAG memoirs*, 2(29-46): 209.

Tursynbek, N.; Petiushko, A.; and Oseledets, I. 2020. Geometry-Inspired Top-k Adversarial Perturbations. *arXiv preprint arXiv:2006.15669*.

Wu, L.; Wang, D.; and Liu, Q. 2019. Splitting Steepest Descent for Growing Neural Architectures. *Advances in Neural Information Processing Systems*, 32: 10656–10666.

Wu, W.; Su, Y.; Chen, X.; Zhao, S.; King, I.; Lyu, M. R.; and Tai, Y.-W. 2020. Towards Global Explanations of Convolutional Neural Networks With Concept Attribution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 8652–8661.

Yeh, C.-K.; Hsieh, C.-Y.; Suggala, A. S.; Inouye, D. I.; and Ravikumar, P. 2019a. On the (in) fidelity and sensitivity for explanations. *arXiv preprint arXiv:1901.09392*.

Yeh, C.-K.; Kim, B.; Arik, S. O.; Li, C.-L.; Pfister, T.; and Ravikumar, P. 2019b. On completeness-aware concept-based explanations in deep neural networks. *arXiv preprint arXiv:1910.07969*.

Zhang, X.; Wang, N.; Shen, H.; Ji, S.; Luo, X.; and Wang, T. 2020. Interpretable deep learning under fire. In *29th {USENIX} Security Symposium ({USENIX} Security 20)*.

Zhang, Z.; and Wu, T. 2020. Learning Ordered Top-k Adversarial Attacks via Adversarial Distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 776–777.

Zhou, Z.; Hooker, G.; and Wang, F. 2021. S-LIME: Stabilized-LIME for Model Explanation. *arXiv preprint arXiv:2106.07875*.