Knowledge and Information Systems (2022) 64:3197–3234
https://doi.org/10.1007/s10115-022-01756-8

SURVEY PAPER

Interpretable deep learning: interpretation, interpretability, trustworthiness, and beyond

Xuhong Li[1] · Haoyi Xiong[1] · Xingjian Li[1] · Xuanyu Wu[2] · Xiao Zhang[3] · Ji Liu[1] · Jiang Bian[1] · Dejing Dou[1,4]

Received: 18 March 2021 / Revised: 23 August 2022 / Accepted: 27 August 2022 /
Published online: 14 September 2022

Abstract
Deep neural networks have been well-known for their superb handling of various machine learning and artificial intelligence tasks. However, due to their over-parameterized black-box nature, it is often difficult to understand the prediction results of deep models. In recent years, many interpretation tools have been proposed to explain or reveal how deep models make decisions. In this paper, we review this line of research and try to make a comprehensive survey. Specifically, we first introduce and clarify two basic concepts—interpretations and interpretability—that people usually get confused about. To address the research efforts in interpretations, we elaborate the designs of a number of interpretation algorithms, from different perspectives, by proposing a new taxonomy. Then, to understand the interpretation results, we also survey the performance metrics for evaluating interpretation algorithms. Further, we summarize the current works in evaluating models' interpretability using "trustworthy" interpretation algorithms. Finally, we review and discuss the connections between deep models' interpretations and other factors, such as adversarial robustness and learning from interpretations, and we introduce several open-source libraries for interpretation algorithms and evaluation approaches.

Keywords Interpretation · Interpretability · Trustworthiness · Interpretable deep learning

1 Introduction

Deep learning models [98] have achieved remarkable performance in a variety of tasks, from visual recognition, natural language processing, reinforcement learning to recommendation systems, where deep models have produced results comparable to and in some cases superior to human experts. Due to their nature of over-parameterization (involving more than millions of parameters and stacked with more than hundreds of layers), it is often difficult to understand

✉ Dejing Dou
doudejing@baidu.com ; dou@cs.uoregon.edu

Extended author information available on the last page of the article

the prediction results of deep models [47]. Explaining[1] their behaviors remains challenging because of their hierarchical non-linearity in a black-box fashion. The lack of interpretability raises a severe issue about the trust of deep models in high-stakes prediction applications, such as autonomous driving, healthcare, criminal justice, and financial services [29]. While many interpretation tools have been proposed to explain or reveal the ways that deep models make decisions, nonetheless, either from a scientific view or a social aspect, explaining the behaviors of deep models is still in progress. In this paper, instead of focusing on the social impacts, regulations, and laws related to deep model interpretations, we would like to focus on the research field by clarifying the research objectives and reviewing the methods proposed.

Interpretation versus Interpretability In this work, we first clarify two concepts that should be distinguished: *interpretations* and *model interpretability*. Interpretations are also named as explanations or attributions that are calculated by *interpretation algorithms* to explain or reveal the ways that deep models make decisions, such as the indication of discriminative features used for model decisions [137], or the importance of every training sample as the contribution for inference [91]. On the other hand, the model interpretability refers to the intrinsic properties of a deep model measuring *in which degree the inference result of the deep model is predictable or understandable to human beings* [47]. In practice, one could apply the interpretation algorithms of *trustworthiness* (introduced below) to further evaluate the model interpretability through matching the interpretations, i.e., the results from interpretation algorithms for a deep model, with the human-labeled results if available, such as [24]. In this way, the comparison of interpretability becomes possible among different models. More evaluation approaches are reviewed and will be introduced later.

Interpretation algorithms and the taxonomy As there are no formal nor well-agreed definitions about the way to interpret a deep model, the interpretation algorithms are usually designed with different principles, such as

- To highlight the important parts of input features on which the deep model mainly relies, using gradients [155], perturbations [55], proxy explainable models [137] and other methods;
- To investigate the inside of deep models to understand the rationale of how models make decisions by visualizing the intermediate features [191, 197], or putting the counterfactual examples to investigate the changes [64];
- To analyze the training data by assessing their individual contributions [91], estimating their learning difficulty [21] or detecting mislabeled samples [128].

This paper reviews the recent interpretation algorithms and proposes a novel taxonomy for categorizing the interpretation algorithms. In brief, the proposed taxonomy has three orthogonal dimensions – (1) *representations of interpretations*, e.g., the input feature importance or the training samples' influences; (2) *the type of the targeting model that the algorithm can be used for*, e.g., differentiable models, models containing specific architectures or other properties; and (3) *relations between interpretation algorithms and the deep model*, e.g., the closed-form expression or the composition of the model. Recent interpretation algorithms can all be categorized to the proposed three-dimensional taxonomy, which will be presented in detail in Sect. 3.

Evaluations on trustworthiness of interpretation algorithms and model interpretability There are two evaluations: one on the trustworthiness of interpretation algorithms, another on the model interpretability.

[1] The subtle differences among *interpretation*, *explanation*, and *attribution* are not considered in this paper, and we use them interchangeably.

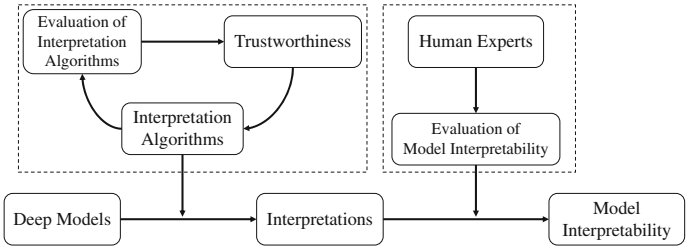


Fig. 1 Scheme about interpretations, interpretation algorithms, trustworthiness, model interpretability and the corresponding evaluations

From previous reviews and outlooks for the interpretations [29, 47, 81, 107, 144], we summarize the most important desiderata for the interpretation algorithms, i.e., the **trustworthiness**. The "trustworthiness" here refers to that the interpretation results are reliable/faithful to arbitrary deep models. That is to say: The trustworthy interpretation algorithm produces the explanations that loyally reveal the model's behaviors, instead of giving results that are irrelevant or just those desired by humans. Incorporating a trustworthy interpretation algorithm, the evaluations on the model interpretability are then meaningful. In Fig. 1, we illustrate the connections between these key concepts and further elaborate these concepts in Sect. 2.

The trustworthiness of the interpretation algorithms could be assessed by designed evaluation approaches for assuring the uses of interpretations, and the interpretability of deep models could be evaluated and measured for identifying the most interpretable ones. Both evaluations have challenges remaining, introduced below.

- Quantifying the utility of trustworthiness of interpretation algorithms is challenging due to the lack of a proper definition of this quantity and well-defined metrics. Though trustworthiness can be understood subjectively that the trustworthy algorithm produces loyal interpretations to the model, the optimal metric is still under study. Simple metrics such as accuracy, precision, and recall are not applicable here.
- The difficulty of evaluating the model interpretability mainly comes from the lack of the *ground truth*. We could not casually annotate "true" interpretations as annotating image labels because interpretation labels might not exist in most cases, or it would be out of objectiveness. Furthermore, obtaining human labeled ground truth for interpretation is labor/time-consuming, which is not scalable over large datasets.

Even in this complex and difficult situation, several efficient and effective approaches have been proposed to evaluate the trustworthiness of interpretation algorithms and model interpretability. The former is mainly based on perturbation evaluations [70, 127, 143] or proxy models [9, 183], while the latter based on expert ground truths [24] or cross-model explanations [104]. In Sect. 4, we comprehensively review the evaluation approaches on both the trustworthiness of interpretation algorithms and model interpretability.

Overview We describe the organization of this survey paper: We introduce the key concepts, including the interpretation algorithm, interpretations, model interpretability, and their relations in Sect. 2. We present the proposed taxonomy for interpretation algorithms and introduce the algorithms accordingly in Sect. 3. Evaluations on the trustworthiness of interpretation algorithms and the model interpretability are introduced in Sect. 4. Section 5 discusses the connections between interpretations and other research topics. Finally, we introduce several open-source libraries for interpretations and related in Sect. 6.

2 Main concepts: interpretations and interpretability

The fuzziness of main concepts *interpretation* and *interpretability* leads to a lot of confusions and hinders the academic process. In this section, we make our efforts to clarify these fuzzy research targets and introduce the definitions of *interpretations*, *interpretation algorithms* and *model interpretability*, with involving the notion of *trustworthiness*.

2.1 Interpretation algorithms and trustworthiness

We first introduce interpretation algorithms. A deep model needs interpretations because the inference output of the model does not show the reasoning inside. An interpretation algorithm is thus designed to produce interpretations to explain the model's decisions and gain insight into its internals of reasoning and rationale. As mentioned previously, there are no formal nor well-agreed definitions about the way to interpret a deep model. We, therefore, adopt a very loose definition about the interpretation: *All the outcomes produced by the interpretation algorithms that help to understand the model are considered as interpretations*.

Instead of directly discussing the interpretations, we introduce the categories of the interpretation algorithms, as they give different information to help humans to understand the deep models. For example, an algorithm obtaining the training samples' learning difficulties helps to inspect the model's training process; An algorithm computing the feature importance helps to realize the most important features that the model uses to make decisions; an algorithm investigating the intermediate results of a neural network helps understand the model's decision-making process. We show a novel taxonomy to fully categorize the existing and potential algorithms and review the corresponding algorithms in Sect. 3.

The interpretation can then lead to the discussion that the model is interpretable or not. However, before that discussion, we should guarantee at the first step that the interpretation algorithm is **trustworthy** and the interpretation can be trusted. The notion of **trustworthiness** is proposed to cover the most important desiderata from the previous review works [29, 107, 122], and can be defined as follows:

– *An interpretation algorithm is trustworthy if it properly reveals the underlying rationale of a model making decisions.*

In this definition, the *underlying rationale* covers all categories of information that help to understand the model, e.g., how the model makes decisions, or the reasoning behind the model making decisions. The word *properly* here targets the issue that the intrinsic underlying rationale behind the model is usually given by an extrinsic algorithm. Extrinsic algorithms may not be part of the targeting model to be interpreted. That is to say, as an additional module to diagnose the model, the interpretation algorithm is at risk of giving explanations that are independent of the model. A sanity check [3] was performed to inspect several gradient-based interpretation algorithms by randomizing parts of parameters in the model and showing the interpretation changes. However, a few algorithms always produce the same interpretations, despite the significant changes of the parameters. Trustworthiness is defined to recover the rationale of the model, whether the model makes the correct decisions or not, instead of yielding information that is independent of the model. Though the definition of trustworthiness is not mathematically rigorous, the idea behind is clear. There are also several evaluations for assessing the trustworthiness, which will be introduced in Sect. 4.1.

Trustworthiness of different interpretation algorithms Due to the differences in representation of explanation results and type of models to be interpreted, the amount of information exposed by interpretation algorithms may be different. Trustworthiness is only required for

the explained information. It would be easy for achieving the trustworthiness if one algorithm explains only a bit of information about the deep model, but this would be rarely useful for any explanation. The trustworthiness is thus an *ad hoc* requirement with respect to the interpretation algorithm and defined to guarantee the information provided by the interpretation algorithm can be trusted.

Relation to self-interpretable models To complete the discussions of trustworthy interpretation algorithms, we note that many researchers are working on effective *self-interpretable models*, to name a few, Capsule Models [73, 142], Neural Additive Models [5] and CALM [88]. We consider this is a particular case within our discussion that the self-interpretable models contain both the model and the intrinsic interpretation algorithm. To be more accurate, the self-interpretable models consist of an intrinsic interpretation algorithm. Moreover, if the model makes decisions based on the intrinsic interpretations, then this interpretation component is without doubt trustworthy.

Fully-interpretable models We also discuss *fully interpretable models* here to get a better understanding about the interpretations and the trustworthiness of interpretation algorithms for black-box deep models. We informally give the definition that a model is fully interpretable if the model is totally understandable by humans. The following models are considered as fully interpretable without too much controversy[2]: a set of limited number of rules; a depth-limited decision tree; a sparse linear model.

Comparison to fully and self-interpretable models To compare across fully interpretable models, self-interpretable models and black-box deep models, we can see: (1) fully interpretable models can be totally understood by showing themselves. (2) Self-interpretable ones can provide explanations with an amount of information by an intrinsic interpretation algorithm. The interpretation algorithms for both fully and self interpretation models are trustworthy. (3) For black-box deep models, it is hard to provide such interpretations and much harder to guarantee the interpretation algorithms be trustworthy. Fortunately, the interpretations may be different and do not provide the fully interpretable explanation results. The trustworthiness only guarantees that the amount of information provided by the interpretation algorithm is correct.

2.2 Model interpretability

From industrial demands, the model interpretability is sometimes more important than other metrics such as accuracy because of safety and social issues in domains of autonomous driving, healthcare, criminal justice, financial services and many others. Though no mathematical definition has been proposed, general agreement about the expression proposed by [47] has been reached. We reclaim their definition of model interpretability as follows.

– *The model interpretability is the ability (of the model) to explain or to present in understandable terms to a human.*

According to other review works [29, 115], "the interpretability of a model is higher if it is easier for a person to reason and trace back why a prediction was made by the model. Comparatively, a model is more interpretable than another model if the prior's decisions are easier to understand than the decisions of the latter".

From the definition of the model interpretability, the expression *understandable to a human* is a subjective notion. It is human-centered [47, 94], making it complicated to target this

[2] Without any limits, even a rule-based model may be too complex for a human to understand the model [107, 141]. This is also the motivation of several works that pursue the sparsity of explanation results [137].

research problem of quantitatively measuring and comparing the interpretability of various models. Till recently, there are not many metrics for quantifying the model interpretability, and Sect. 4.2 will introduce the existing evaluation approaches on the model interpretability.

We give an intuitive example to show that different models may have different interpretability. Take image classification [43, 175] as the task, and a trustworthy algorithm of analyzing the input-output relations as the interpretation algorithm. We consider two models, and the produced interpretations locate different image pixels. It is easier to understand if the interpretation aligns with the object parts in the image, while it is harder to understand if the interpretation locates at the background or another accompanied object in the image for recognizing the target object. Although the trustworthy algorithm reveals the rationales of both models, we prefer the former model because its way of making decisions is more direct to human understandings.

2.3 Toward interpretable deep learning

This section defined the trustworthiness of interpretation algorithms and the model interpretability. We emphasize several points that usually confuse the field with more explicit remarks.

Interpretation algorithms, interpretations and model interpretability The notions of interpretation algorithms, interpretations, and model interpretability should be distinguished. Only the interpretability among all these expressions is a property of the model. Interpretation algorithms are designed to analyze the black-box model. Algorithms must be trustworthy; otherwise, the interpretations do not reveal the model's internals. Their relations and differences are illustrated in Fig. 1.

Summary of desiderata for interpretations In this section, the proposed desiderata is the trustworthiness for interpretation algorithms. Researchers [29, 47, 81, 101, 107, 183] also proposed many other desiderata for interpretations, interpretation algorithms or interpretability, such as fairness, privacy, reliability, robustness, causality, trust, fidelity, faithfulness, transferability, informativeness, transparency, plausibility, satisfaction, accountability, etc. However, we note that (1) properties, such as informativeness, plausibility, satisfaction, refer to whether the interpretation is understandable to humans, and are different from the trustworthiness in this paper that refers to algorithms; (2) properties, such as reliability, robustness, trust, fidelity, faithfulness, transparency, are similar to trustworthiness or can be comprised by the general definition of trustworthiness; (3) properties, such as causality, transparency, depend on the *underlying rationale* in our context; (4) properties, such as fairness, transferability, privacy, are the standards to constrain the models; and (5) others (e.g., accountability and traceability) are more related to holistic evaluations of the systems. There is slight difference and specific requirements in various scenarios, but the proposed trustworthiness is only for interpretation algorithms.

Deep models for high-dimensional data for scientific discovery Though the motivation of interpretations and interpretability at the beginning is to help humans understand the deep models, the interpretations sometimes lead to other valuable and promising findings. Deep models may be more efficient than humans to cope with high-dimensional data. From molecules [84, 133] to black holes [86], from chemistry [62] to games [152], deep models could be used to solve many problems. However, without interpretations, the knowledge discovered by deep models is still unknown for humans, or the scores obtained are not semantic and not fully understood by humans. Interpretations in these cases could be helpful to find new intelligent patterns and discover new scientific theories. For example, from a

perspective of rationale processes, interpretations can help humans to understand how a model infers; Or a feature analysis algorithm can help to identify the most important features that the model uses; Or a tool of investigating the data can help find the typical data samples or the most influential ones that explain how the model makes decisions. These algorithms are all included in this paper and will be discussed in the following section.

3 Interpretation algorithms: taxonomy, algorithm designs, and miscellaneous

This section introduces the interpretation algorithms in recent years, with a proposed taxonomy of three dimensions. For each algorithm, we give a brief introduction and follow the taxonomy for the categorization. A discussion is also provided for future works at the end of this section.

3.1 Taxonomy

We categorize the existing interpretation algorithms according to three orthogonal dimensions: representations of interpretations, targeting model's types for interpretations, and the relation between interpretation algorithms and models. We list the options in each dimension for a better comparison.

For different applications and interpretation requirements, the representations of interpretation are various:

- **Feature (Importance)**. These algorithms aim at estimating the feature importance/contribution with respect to the final objective. This includes the analyses on the dimensions of input raw data and extracted features, e.g., images, texts, audios etc.; and intermediate features inside models, e.g., the activations of neural networks; or latent features in GANs.
- **Model Response**. Algorithms here generally propose to generate or find new examples and see the model's responses, so as to investigate the model behaviors on certain patterns, prototypes, or discriminative features by which the model makes decisions.
- **Model Rationale Process**. Though deep models are complex, they can be substituted by interpretable models, to gain insights on the rational process inside. Algorithms here interpret the deep model by indicating the path that the model makes decisions.
- **Dataset**. Instead of interpreting deep models, algorithms here propose to explain the data samples in the training set by showing how they affect the optimization phase of deep models.

Interpretation algorithms cope with different types of models:

- **Model-agnostic**. Algorithms are included here that completely consider the models as black boxes and do not investigate the inside of models.
- **Differentiable model**. This subset of algorithms contains only algorithms that address the interpretations of differentiable models, especially neural networks. Note that model-agnostic algorithms also cover this subset.
- **Specific model**. This family of algorithms can only be applied to certain types of models, e.g., convolutional neural networks (CNNs), generative adversarial networks (GANs), Graph Neural Networks (GNNs). This is a narrower family than the previous one.

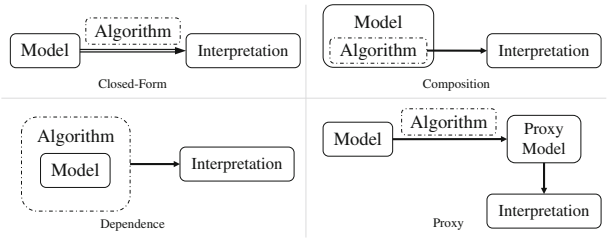


Fig. 2 Illustration of relations between the interpretation algorithm and the model. Four relations are illustrated: closed-form, composition, dependence and proxy

The third dimension for categorizing interpretation algorithms is the relation between the interpretation algorithm and the model:

- **Closed-form**. These algorithms derive a closed-form formula from the target model and output interpretable terms.
- **Composition**: Algorithms here can be considered as components of (interpretable) models, usually obtained during training.
- **Dependence**: These algorithms build new operations upon the target model after training and output interpretable terms.
- **Proxy**. Unlike dependence, algorithms here obtain, via learning or derivation, a proxy model for explaining the behavior of models.

For a better illustration, four of relations between interpretation algorithms and deep models are shown in Fig. 2.

We have introduced the proposed taxonomy of three dimensions: Representation, Model Type and the Relation. In the following subsection, we will present most of the recent interpretation algorithms. We also give a categorization of all these algorithms with respect to the proposed taxonomy in Table 1.

3.2 Interpretation algorithms

LIME and model-agnostic algorithms LIME [137] presents a locally faithful explanation by fitting a set of perturbed samples near the target sample using a potentially interpretable model, such as linear models and decision trees. We define a model $g \in G$, where G is a class of interpretable models. The domain of g is $\{0, 1\}^{d'}$ and its complexity measure is $\Omega(g)$. Let $f : \mathbb{R}^d \to \mathbb{R}$ be the model being explained and $\pi_x(z)$ be the proximity measure between a perturbed sample z and x. Finally, let $L(f, g, \pi_x)$ be a measure of the unfaithfulness of g in approximating f in the locality defined by π_x. LIME produces explanations by the following:

$$\xi(x) = \arg\min_{g \in G} L(f, g, \pi_x) + \Omega(g). \tag{1}$$

The obtained explanation $\xi(x)$ interprets the target sample x, with linear weights when g is a linear model. LIME is model-agnostic, meaning that the obtained proxy model is suitable for any model. Similarly, several model-agnostic algorithms, such as Anchors [138], SHAP [110], RISE [127], MAPLE [130], target interpreting features and provide feature importance or contributions to the final decision.

Global interpretation algorithms Feature importance analysis is a common tool for explaining the model outputs with respect to inputs. The aforementioned approaches can

be categorized into feature importance analysis, while their interpretations are for individual examples, giving unique results for each different example. Different from these "local" interpretations, "global" interpretations provide feature importance in an overall vision of the model. Global interpretations for deep models are usually based on local ones, and an aggregation of local interpretations is performed to obtain the global feature importance, while the difference resides in the aggregation approach, e.g., LIME-SP [137], NormLIME [6] and GALE [166].

Input gradient-based algorithms The input gradient attributes the important features in the input domain. However, for deep nonlinear models with numerous layers stacked, the gradients would be vanished or saturated during the back-propagation and thus contain noises.

SmoothGrad [155] proposed to remove the noises by averaging the gradients of a number of noised inputs. We take visual tasks as an example: Given input image x, neural networks compute a class activation function S_c for class $c \in C$. A sensitivity map can be constructed by calculating the gradient of M_c with respect to input x: $M_c(x) = \partial S_c(x)/\partial x$. However, the saliency maps are often noisy because of sharp fluctuations of the derivative. To smooth the gradients, multiple Gaussian noises are added to the input image, and the saliency maps are averaged. SmoothGrad is defined as follows:

$$\hat{M}_c(x) = \frac{1}{n}\sum_{1}^{n} M_c(x + \mathcal{N}(0,\, \sigma^2)). \tag{2}$$

Integrated Gradient (IG) [160] aggregates the gradients along with the inputs that lie on the straight line between the baseline and input. Let F be a neural network, x be the input, and x' be the baseline input, which can be a black image for computer vision models and a vector of zeros for word embedding in text models. The integrated gradients along the ith dimension is

$$\mathrm{IG}_i(x) = (x_i - x_i') \times \int_{\alpha=0}^{1} \frac{\partial F(x' + \alpha \times (x - x'))}{\partial x_i} \mathrm{d}\alpha. \tag{3}$$

An axiom called *completeness* is satisfied, which states that the attributions add up to the difference between the output of F at input x and baseline x'.

Other input gradient-based algorithms include DeepLIFT [150], VarGrad [3], GradSHAP [110], and FullGrad [156].

Layer-wise relevance propagation Layer-wise relevance propagation (LRP) [16] is also an input feature attribution algorithm. Instead of using proxy models, perturbations or gradients, LRP recursively computes a Relevance score for each neuron of layers, so as to understand the contribution of a single pixel of an image x to the prediction function $f(x)$ in an image classification task.

$$f(x) = \cdots = \sum_{d=1}^{V^{(l+1)}} R_d^{(l+1)} = \sum_{d=1}^{V^{(l)}} R_d^{(l)} = \cdots = \sum_{d=1}^{V^{(1)}} R_d^{(1)}, \tag{4}$$

where $R_d^{(l)}$ is the Relevance score of the dth neuron at the lth layer, $V^{(l)}$ indicates the dimension of lth layer, and $V^{(1)}$ is the number of pixels in the input image. Iterating Eq. (4) from the last layer, which is the classifier output $f(x)$ to the input layer x consisting of image pixels, then yields the contribution of pixels to the prediction results. Based on the idea of back-propagating Relevance scores, LRP can be extended to other neural networks, even with special and complex nonlinear operations [27, 118]. To adapt LRP to specific tasks, many

variants have been proposed, such as Contrastive LRP [67] which produces pixel-wise explanations of instance objects, Softmax-Gradient LRP [79] which gives explanations focusing on discriminating possible objects in the images, and Relative Attributing Propagation (RAP) [123] which focuses on both positive and negative features. Furthermore, extended LRPs [34, 169] can be helpful to interpret Transformer models [45, 48, 159].

CAM and variants Given a CNN and an image classification task, classification activation map (CAM) [197] can be derived from the operations at the last layers of the CNN model and show the important regions that affect model decisions. Specifically, for a given category c, we expect the unit corresponding to a pattern of the category in the receptive field be activated in the feature map. The weights in the classifier indicate the importance of each feature map in classifying category c. Therefore, a weighted sum of visual patterns illustrates the important regions of a category. Let $f_k(x, y)$ denote the activation of unit k in the last convolutional layer at spatial location (x, y), $F_k = \sum_{x,y} f_k(x, y)$ be the global average pooling for unit k, and w_k^c be the weight corresponding to class c for unit k so that $\sum_k w_k^c F_k$ is the input to softmax for class c. Then the activation map for class c is:

$$M_c(x, y) = \sum_k w_k^c f_k(x, y). \tag{5}$$

GradCAM [145] further looks at the gradients flowing into the convolutional layer to give weight to activation maps. Let y^c be the score for class c before the softmax, A^k be feature map activations of the unit k in a convolutional layer, the neuron importance weight α_k^c is the global-average-pooled gradient of y^c with respect to A^k:

$$\alpha_k^c = \frac{1}{Z} \sum_i \sum_j \frac{\partial y^c}{\partial A_{i,j}^k}. \tag{6}$$

The localization map is a weighted combination of activation maps:

$$L_{Grad-CAM}^c = ReLU(\sum_k \alpha_k^c A^k). \tag{7}$$

ScoreCAM [174] also uses gradient information but assigns importance to each activation map by the notion of *Increase of Confidence*. Given an image model $Y = f(X)$ that takes in image X and outputs logits Y. The kth channel of convolutional layer l is denoted A_l^k. With baseline image X_b and category c, the contribution A_l^k toward Y is:

$$C(A_l^k) = f^c(X \circ H_l^k) - f^c(X_b), \tag{8}$$

where $H_l^k = s(Up(A_l^k))$. $Up(\cdot)$ is the operation that upsamples A_l^k into the input size and s normalizes each element into [0, 1]. ScoreCAM is defined as:

$$L_{Score-CAM}^c = ReLU(\sum_k \alpha_k^c A_l^k), \tag{9}$$

where $\alpha_k^c = C(A_l^k)$.

More CAM variants have been recently proposed, e.g., GradCAM++ [32], CBAM [178], Respond-CAM [196], and Ablation-CAM [44].

Perturbation-based algorithms To investigate important features in the input, a straightforward way is to measure the effect of perturbations applied to the input [54, 55]. This idea is quite simple: The random perturbations on the features would lead to different changes in the model's predictions, where larger changes would be observed for more important features.

Note that perturbation can be also used for evaluating the trustworthiness of interpretation algorithms when we are not aware of interpretation ground truth [143, 172].

Counterfactual examples Using counterfactual examples to explain the model behaviors is also an important direction for understanding the black boxes. Generally, the counterfactual examples have changes in the input that are as small as possible, but would completely change the decision made by the model. The changes in input would be a clue for explaining the model's behavior. Most counterfactual-example approaches, such as FIDO [31], DiCE [121], and several others [64, 97], to generating counterfactual examples are based on the optimization with sparsity constraints or toward the smallest changes in input. Using counterfactual examples to explain the model behaviors can also be included in causal inference [126], which is considered as a new perspective for model interpretability [120, 179]. Detailed reviews on counterfactual explanations can be found in [12, 167, 173].

Adversarial examples Adversarial examples are very related to counterfactual ones with similar optimization methods, while adversarial examples are used to reveal the vulnerability of the deep model and often attack the AI systems. Adversarial examples in vision tasks are usually the imperceptible changes in the images which mislead the model's decision. Note that analyses on the adversarial examples [58, 77] show the connections to the understanding of the deep learning process and robustness of the trained deep model.

TCAV Given a set of examples representing a concept of human interest (such as an object, a pattern, a color etc.), TCAV [87] seeks a vector in the space of activations at some layer to represent this concept. Precisely, by defining a concept activation vector (or CAV) as the normal to a hyperplane, TCVA separates examples according to the existence of this concept in the activations: Given one example in a particular class, along the direction of a CAV, the directional derivative of this example contributes a score if it is positive, and the ratio of examples that have positive directional derivatives over all examples in this class is defined as the TCAV score. CAV finds examples of a semantic concept learned by the intermediate layers of a deep model, contributing to the predictions while TCAV quantitatively measures the contributions of this concept.

Prototype To explain the classification models, finding the typical exemplar for each category is also effective and direct. Humans can understand better that the model identifies the featured prototype to make decisions. Chen et al. [35] proposed ProtoPNet, which explains the deep model by finding prototypical parts of predicted objects and gathering evidence from the prototypes to make final decisions. Another method named ABELE [69] generates exemplar and counter-exemplar images, labeled with the class identical to, and different from, the class of the image to explain, with a saliency map, highlighting the importance of the areas of the image contributing to its classification.

As a technique for generating prototypes, activation maximization generally computes the prototypes through an optimization process:

$$\max_{\boldsymbol{x}} \ \log p(y_c|\boldsymbol{x}) - \lambda \|\boldsymbol{x}\|^2, \tag{10}$$

where $p(y_c|\boldsymbol{x})$ is the probability given by a deep model with $\boldsymbol{x}$ as input, and the second term is the constraint for generating the prototype. However, the constraint can be replaced by many other choices [49, 113, 124, 153]. A tutorial for this direction is cited [119]. More works related to prototypes or exemplars for interpretations can be found in [23, 26, 103, 116].

Proxy models for rationale process The reasoning process or the underlying rationale of deep models is complex due to the nonlinearity and enormous computations. It is difficult for humans to know the exact steps of the rationale process with semantics inside the black

boxes. However, this rationale process can be proxied by graph models [190] or decision trees [192], which provide a decision-making path that is more interpretable to humans. Moreover, deep neural networks can be combined with decision forest models [92] or distilled into a soft decision tree [57]. A model-agnostic approach for interpreting rationale process named BETA [96] allows to learn (with optimality guarantees) a small number of compact decision sets, each of which explains the behavior of the black box model in specific, well-defined regions of feature space.

Forgetting events Forgetting events are defined by [164] for analyzing the training examples using training dynamics. Given a dataset $D = (x_i, y_i)_i$, after t steps of SGD, example x_i undergoes a forgetting event if it is misclassified at step $t+1$ after having been correctly classified at step t. Forgetting events signify samples' interactions with decision boundaries, and the samples play a part equivalent to support vectors in the *support vector machine* paradigm. Unforgettable examples are samples learnt at step $t^* < \infty$ and never misclassified for all $k \geq t^*$. They are easily recognizable samples that contain obvious class attributes. Whereas examples with the most forgetting events are ambiguous without clear characteristics of a certain class, and some are noisy samples.

Dataset cartography Dataset cartography [161] looks into two measures for each sample during the training process—the model's confidence in the true class and the variability of confidence across epochs. Therefore, training examples can be categorized as easy-to-learn, hard-to-learn, or ambiguous based on their position in the two-dimensional map. Consider training dataset $D = (x, y^*)_{i=1}^N$ where x_i is the ith sample and y_i^* is the true label. After training for E epochs, the confidence is defined as the mean probability of true label across epochs:

$$\hat{\mu_i} = \frac{1}{E} \sum_{e=1}^{E} p_{\theta^{(e)}}(y_i^* | x_i), \tag{11}$$

where $p_{\theta^{(e)}}$ is the probability with parameters $\theta^{(e)}$ at the end of the eth epoch. The variability is the standard deviation of $p_{\theta^{(e)}}(y_i^* | x_i)$:

$$\hat{\sigma_i} = \sqrt{\frac{\sum_{e=1}^{E} (p_{\theta^{(e)}}(y_i^* | x_i) - \hat{\mu_i})^2}{E}}, \tag{12}$$

AUM Another method for analyzing the training dynamics is proposed to compute the area under the margin (AUM) [128]:

$$\text{AUM}(\boldsymbol{x}, y) = \frac{1}{T} \sum_{t=1}^{T} (z_y^{(t)}(\boldsymbol{x}) - \max_{i \neq y} z_i^{(t)}(\boldsymbol{x})), \tag{13}$$

where $z_i^{(t)}(\boldsymbol{x})$ is the logit, computed by the model, of ith class at tth epoch during training with respect to the example $\boldsymbol{x}$.

Influence functions Influence functions [91] identify the training samples most responsible for a model prediction by upweighting a sample by some small value and analyze its effect on the parameters and the loss of the target sample. Given input space X and output space Y, we have training data $z_1, \ldots, z_n$, where $z_i = (x_i, y_i) \in X \times Y$. Let $L(z, \theta)$ be the loss where $\theta \in \Theta$ are the parameters. The optimal $\hat{\theta}$ is given by $\hat{\theta} = argmin_{\theta \in \Theta} \frac{1}{n} \sum_{i=1}^{n} L(z_i, \theta)$. The influence of upweighting training point z on the loss at the test point z_{test} is:

$$I_{up,loss}(z, z_{test}) = -\nabla_\theta L(z_{test}, \hat{\theta})^T H_{\hat{\theta}}^{-1} \nabla_\theta L(z, \hat{\theta}), \tag{14}$$

where $H_{\hat{\theta}} = \frac{1}{n}\sum_{i=1}^{n} \nabla_{\theta}^{2} L(z_i, \hat{\theta})$. Based on influence functions, several techniques [38, 90] have been proposed with improvement.

Contributions of long-tailed training examples Instead of identifying mislabeled samples, easy/difficult-to-learn samples from the training set, more theoretical works on detecting the long-tail examples and outliers [28, 52, 53]. Most of them investigate the connections between the memorization capacity of deep models [187] and the learning process, in order to know the contributions of training examples, including long-tailed ones and outliers.

Interpretations on GNNs Graph Neural Networks (GNNs) are a powerful tool for learning tasks on structured graph data. Like other deep learning models, GNNs show the black-box fashion and are required to explain their prediction results and rationale processes. Without requiring modification of the underlying GNN architecture, GNNExplainer [184] leverages the recursive neighborhood-aggregation scheme to identify important graph pathways as well as highlight relevant node feature information that is passed along edges of the pathways. Recently, more researches focus on the interpretations of GNN models, such as GraphLIME [76], CoGE [51], Counterfactual explanations on GNNs [18] and others [20, 111, 132].

Interpretations on GANs Generative adversarial networks (GANs) are a popular generative model based on two adversarial networks, where one generates synthesized examples, and another tries to classify generated examples from natural examples. Interpretations on GANs mainly search for semantically meaningful directions. Compared with labeled semantics, Bau et al. [25] proposed GAN dissection to find semantic neurons in generative models and modify the semantics in the generated images. Instead of relying on labels, Voynov et al. [171] found semantically meaningful directions in an unsupervised way from the intermediate layers of generative models. Similarly, Shen et al. [149] proposed a closed-form factorization method for identifying semantic neurons. Note that there are other methods for explaining the generative models [131, 170, 180].

Information flow In some deep learning models, there are multiplicative scalar weights that control information flow in some parts of a network. The most common examples are attention [17] and gating:

$$c^{att} = \sum_{i} \alpha_i^{att} h_i, \qquad c^{gate} = \alpha^{gate} h \tag{15}$$

The attention weights α^{att} ($\sum_i \alpha_i^{att} = 1$) and the gate values α^{gate} ($\alpha^{gate} \in [0, 1]$) are usually interpretable because their values represent the strength of the corresponding information pathways. Attention and gating are frequently used in NLP models, and there have been plenty of works aiming to understand the model through these weights, such as Rollout [2], Seq2Seq-Vis [157] and others [61, 158], or to investigate the reliability of using them as explanations [82, 148, 177]. As well, these ideas have also been used in Vision Transformers [48] for explaining image classification models [34, 185] or bi-modal transformer models [33].

Self-generated explanations Using text generation techniques, a model can explicitly generate human-readable explanations for its own decision. A joint output-explanation model is trained to produce an prediction and simultaneously generate an explanation for the reason of that prediction [14, 93, 109]. This requires some kind of supervision available to train the explanation part of the model.

Inductive biases toward interpretation modules Different from post-hoc explanations after the optimization process, some works focus on designing inductive biases during training to encourage the model to be more interpretable. By simple abstraction, the objective function for this purpose can be written as

Table 1 Categorization of interpretation algorithms with respect to the proposed taxonomy

Algorithms	Representation	Model type	Relation
LIME and variants	Feature	Model-Agnostic	Proxy
Global interpretation	Feature	Model-Agnostic	Proxy
Input-gradient based	Feature	Differentiable	Dependence
LRP and variants	Feature	Differentiable	Dependence
CAM and variants	Feature	Specific (CNNs) or Differentiable	Closed-form or dependence
Perturbation-based	Feature	Model-Agnostic	Dependence
Counterfactual examples	Response	Model-Agnostic or Differentiable	Dependence
Adversarial examples	Response	Model-Agnostic or Differentiable	Dependence
TACV	Feature	Differentiable	Proxy
Prototype-based	Response	Model-Agnostic or Differentiable	Proxy
Proxy models for rationale process	Rationale	Specific (CNNs)	Proxy
Training dynamics based	Dataset	Model-Agnostic	Dependence
Influence functions and variants	Dataset	Differentiable	Closed-Form or Dependence
Contributions of training examples	Dataset	Differentiable	Dependence
Interpretations on GNNs	Feature	Specific (GNNs)	Dependence
Interpretations on GANs	Feature	Specific (GANs)	Dependence
Information flow	Feature	Specific (Transformers)	Dependence
Self-generated explanations	Feature	Specific (NLP)	Composition
Self-interpretable models	Rationale	Specific (Self-Interpretable)	Composition

Algorithms are listed following the order of presentation in Sect. 3.2. Note that, each row may contain several algorithms which they may target at explaining different types of models or have different relations to the models. Here the publications in each category of algorithms can be found in the corresponding paragraphs, and are not repeated for a compact table presentation

$$Loss = L(f(x), y) + \alpha R, \tag{16}$$

where $f(x)$ represents the deep model output with x as input, y is the ground truth, L is the loss function, specifically the cross entropy for standard supervised classification problem, and R is the objective function for biasing toward interpretable models. Various approaches [46, 114, 140, 191] have been proposed to improve the interpretability during training. More encouragingly, Sabour et al. [142] designed a self-interpretable deep model where each neuron outputs semantic features.

3.3 Categorization and discussion

We have introduced a large number of typical interpretation algorithms and categorized them according to the proposed taxonomy, so as to provide a clear illustration in this research field. We hope the taxonomy can shed light on future improvements/extensions on explaining

Table 2 List of interpretation algorithm publications

Methods	Publications (non-exhaustive)
LIME and variants	LIME [137], Anchors [138], SHAP [110], RISE [127], MAPLE [130]
Global interpretation	LIME-SP [137], NormLIME [6], GALE [166]
Input-gradient based	SmoothGrad [155], IG [160], DeepLIFT [150], VarGrad [3], Grad-SHAP [110], FullGrad [156]
LRP and variants	LRP [16, 27, 118], Contrastive LRP [67], Softmax-Gradient LRP [79], RAP [123], Chefer et al. [34]
CAM and variants	CAM [197], GradCAM [145], ScoreCAM [174], GradCAM++ [32], CBAM [178], Respond-CAM [196], Ablation-CAM [44]
Perturbation-based	Fong et al. [54, 55], Samek et al. [143], Vu et al. [172],
Counterfactual examples	FIDO [31], DiCE [121], Goyal et al. [64], Laugel et al. [97]
Adversarial examples	Geirhos et al. [58], Ilyas et al. [77]
TACV	TACV [87]
Prototype-based	ProtoPNet [35], ABELE [69]
Proxy models for rationale process	Zhang et al. [190, 192], BETA [96]
Training dynamics based	Forgetting Events [164], Datasets Cartography [161], AUM [128]
Influence functions and variants	Influence Functions [91], Group Influences [90], HYDRA [38]
Contributions of training examples	Carlini et al. [28], Feldman et al. [52, 53]
Interpretations on GNNs	GNN Explainer [184], GraphLIME [76], CoGE [51]
Interpretations on GANs	GAN Dissection [25], Voynov et al. [170, 171], Shen et al. [149]
Information flow	Rollout [2], Seq2Seq-Vis [157], Chefer et al. [33, 34], TAM [185]
Self-generated explanations	Atanasova et al. [14], Kumar et al. [93], Liu et al. [109]
Self-interpretable models	Capsule [73, 142], Neural additive models [5], CALM [88]

Algorithms are listed following the order of presentation in Sect. 3.2

(deep) learning models. We show the categorization of all these algorithms with respect to the proposed taxonomy in Table 1, and gathering interpretation algorithms according to the categorization in Table 2 for a quick glimpse.

Table 1 shows that there are many methods of the **Feature** representation and only a few **Rational** ones; many **Proxy** and **Dependence relations** but a few **Closed-Form**. We argue that both of these observations were due to the challenging analyses of complex deep neural networks. The rationale and the closed-form of deep models are still hard to understand or even approximate. From the categorization, we also would like to point out the blanks that may indicate some unexplored directions for future perspectives. For example, no **Model-Agnostic** algorithms have the **Composition** relation with models. While the input–output sensitivity analysis methods are currently developed, improving the input-output interpretations can be a good perspective. Moreover, we should note that the adversarial attacks do not only aim at trained models [30], but the interpretations [8, 56, 71]. We leave the further investigations for future work.

3.4 Interpretations on specific application domains

We do not explicitly categorize the interpretation algorithms according to their application domains because (1) the algorithm used in one specific domain may also be applicable on a

broader scope with little modifications, especially for model-agnostic algorithms; and (2) for model-specific algorithms, the categorization on the model type generally overlaps with the one on the application domain. For completeness, we discuss recent works of deep model interpretations in the following domains: reinforcement learning, recommendation systems, and medical domains. These applications are slightly different from image classification or sentiment analyses and may require interpretations in a unique form, but most algorithms introduced previously can be used directly.

3.4.1 Deep reinforcement learning (DRL)-related domains

Reinforcement learning (RL) [85] is an area of machine learning concerned with how intelligent agents ought to take actions in an environment in order to maximize the notion of cumulative reward. Deep learning methods have recently enabled RL to decision-making problems that were previously intractable, such as playing games [117, 151, 168] and training robots [10, 99, 100]. DRL is also applicable and shows potentials of application in healthcare, finance and business management [13, 105], where human security and property safety issues should be considered, leading to the demands of explainable RL [134].

According to recent surveys [13, 105], DRL methods are generally based on DNNs to approximate value functions or find policies. Most methods directly learn the objectives from raw inputs, especially for visual tasks where the images are used as inputs for estimating the value functions. For those methods, input feature-related interpretation algorithms, such as LIME and SmoothGrad, have already been explored for explaining DRL methods [15, 65, 80, 135]. However, as we discussed before, interpretation algorithms may expose different amount of information of the deep models, and in some real-world situations, different interpretation algorithms are required. For critical problems concerning human security and property safety, showing the input–output relations of deep models is sometimes not persuadable for consumers. The rationale inside the deep model may be required and has not been much investigated yet in this field.

3.4.2 Recommendation systems

The recommendation system [139] is a subclass of the information retrieval domain that seeks to predict the "rating" or "preference" a user would give to an item. With the growing information available on the Internet, it becomes more and more difficult to find the items of interest by users themselves. For many web applications, the recommendation systems are an essential method for providing a better user experience [193]. Based on all kinds of information provided by users explicitly or implicitly, the recommendation system filters and sorts a list of items of interest in a personalization way.

There are three reasons for explainable recommendation systems. The first one is to gain users' trust in the recommendation system. Explanations help to improve the transparency, persuasiveness and user satisfaction of the recommendation system. The second is to facilitate engineers to debug the recommendation algorithm. Explanations provide analyses how the deep model works, and it would be easy to locate the bugs with explanations. The first two arguments are borrowed from previous reviews [193, 195]. The third is to prevent the privacy and social issues. The recommendations may be computed based on features that have privacy or ethical issues. We would not like to have a recommendation system that may lead to these issues. Explanations can thus be used to expose and prevent this problem.

Classic recommendation methods, including collaborate filtering [19, 72], are interpretable, while the usages of black-box deep models [39, 40, 63] increases the opacity of

recommendation systems. Recent works on explainable recommendation systems can be categorized following our proposed taxonomy, and most of them focus on designing interpretable modules [36, 102, 147, 162]. We refer interested readers to the survey on explainable recommendation systems [195].[3]

3.4.3 Deep learning applications to medical applications

Deep learning methods have been recently applied on medical domains, especially on medical imaging analyses [108], such as the classification of Alzheimer's [83], lung cancer detection [75], tuberculosis diagnosis [136], retinal disease detection [146], etc. Though researchers show the potentials of using deep learning methods in helping the diagnostics, the applications in the real-world situations of healthcare, clinics, hospitals and rehabilitation are very critical, because a single failure would cause irreparable damages. Explanations for deep learning-based methods are more urged in this field than in other fields, to gain the trust of physicians, regulators as well as the patients [154].

Interpretation algorithms proposed in this specific domain have been surveyed [154, 163]. Most of them are aligned with the general ones as reviewed in this work, because the network architectures are the same and the tasks are similar. The difference mainly resides in the data distribution and the domain expert knowledge. Interpretation algorithms are technically applicable and their trustworthiness can be evaluated in medical domains. In spite of the advances, however, currently deep learning-based methods have not achieved a significant deployment in the clinics still due to the lack of interpretability [154]. This indicates that the new interpretation tools are still required in this domain.

4 Trustworthiness evaluations of interpretation algorithms and model interpretability evaluations

Previous section focuses on the interpretation algorithms and interpretation results. This section summarizes the current works in evaluating the trustworthiness of interpretation algorithms, and the deep models' interpretability. To emphasize, the model interpretability is measured based on trustworthy interpretation algorithms. Before introducing model interpretability evaluation, we present the evaluation methods for assuring the trustworthiness of interpretation algorithms in Sect. 4.1. Then, given a trustworthy interpretation algorithm, in Sect. 4.2 we present a few evaluation methods for the interpretability of deep models.

4.1 Trustworthiness evaluations of interpretation algorithm

Perturbation-based evaluations The perturbation-based evaluation of interpretation algorithms follows the intuition that flipping the most salient pixels first should lead to high performance decay. Perturbation-based examples can therefore be used for the trustworthiness evaluations of interpretation algorithms [41, 70, 143, 172]. The main metric MoRF, Most Relevant First, (or LeRF, Least Relevant First, respectively), calculates the area under the curve (AUC), where the curve is of the probabilities predicted by the model after removing most relevant features (or least relevant features respectively). MoRF would drop very quickly at beginning and LeRF would retain at a high value until the end, if the explanation

[3] We also note that whether the usage of deep models improves the recommendation system is an open discussion [42], but this is out of the scope of this survey.

is trustworthy. They are usually used together and both have the same objective of evaluating the trustworthiness of the explanation.

In a different view [55, 74] that "without re-training, it is unclear whether the degradation in model performance comes from the distribution shift or because the features that were removed are truly informative." So Hooker et al. [74] proposed to remove the most important features, extracted by the interpretation algorithms, and then retrain the model, to measure the degradation of model performance and evaluate the trustworthiness of interpretation algorithms. We believe that the prohibitive computation cost added by the retraining step is meaningful for explaining the learning process (how the features/pixels were learned by a specific architecture of models), but contributes less to explain one trained model in a post-hoc way.

Evaluations by randomizing parameters There is no need for retraining in some cases, and we can identify untrustworthy interpretation algorithms by simply randomizing parameters. Adebayo et al. [3] found that even with random weights at the top layers of the network, a number of saliency map-based approaches were still able to locate the important regions of the input images, and proved that these methods do not depend on the models. Adebayo et al. [4] summarized the uses of interpretation algorithms for model debugging, i.e., to detect spurious correlation artifacts (data contamination), diagnose mislabeled training examples (data contamination), differentiate between a (partially) re-initialized model and a trained one (model contamination), and detect out-of-distribution inputs (test-time contamination).

BAM Yang et al. [181] proposed a framework, named Benchmarking Attribution Methods (BAM), for benchmarking interpretation algorithms through a manually created dataset where objects are randomly pasted into images, and a set of models trained on that dataset. BAM carefully generates a semi-natural dataset, where objects are copied into images of scenes, so each image has an object label and a scene label. Then with models trained on this dataset and test examples, a target interpretation algorithm is evaluated by this framework, giving relative importance rankings for input features, which can be validated by ground truth from the generated dataset. The intuition behind BAM is that relative importance has a ground truth ranking, which can be controlled by the crafted dataset and used for comparing with the one given by interpretation methods, and then BAM can quantitatively evaluate the trustworthiness of the algorithm.

Trojaning Model trojaning attacks [37, 68] indicate visual dataset contamination, where a subset of images are modified by giving a specific trigger (e.g., a yellow square is attached to the right bottom of the image) to the desired target. This attack poisons the trained model that the trigger is the only feature for classifying the desired target. Benefit from trojaning attacks, Lin et al. [106] proposed to verify the interpretation algorithm on the trojaned models. The qualified algorithm should highlight pixels around the trigger in contaminated images instead of object parts. Using the triggers as ground truth, Lin et al. [106] evaluated the trustworthiness of interpretation algorithms.

Infidelity and sensitivity The desired properties relating to trustworthiness have been discussed in [9, 183]. We reclaim the two definitions of (in)fidelity and sensitivity, which objectively and quantitatively measure the trustworthiness of interpretation algorithms. Given a black-box function $\boldsymbol{f}$, an interpretation algorithm Φ, a random variable $\boldsymbol{I} \in \mathbb{R}^d$ with probability measure $\mu_{\boldsymbol{I}}$, which represents meaningful perturbations of interest, and a given input neighborhood radius r, the infidelity and sensitivity of Φ of the target interpretation algorithm as:

$$\text{INFD}(\Phi, \boldsymbol{f}, \boldsymbol{x}) = \mathbb{E}_{\boldsymbol{I}\sim\mu_{\boldsymbol{I}}}(\boldsymbol{I}^T \Phi(\boldsymbol{f}, \boldsymbol{x}) - (\boldsymbol{f}(\boldsymbol{x}) - \boldsymbol{f}(\boldsymbol{x} - \boldsymbol{I}))^2), \tag{17}$$

$$\text{SENS}_{\text{MAX}} = \max_{\|\boldsymbol{y}-\boldsymbol{x}\| \leq r} \|\Phi(\boldsymbol{f}, \boldsymbol{y}) - \Phi(\boldsymbol{f}, \boldsymbol{x})\|, \tag{18}$$

where $\boldsymbol{I}$ represents significant perturbations around $\boldsymbol{x}$ and can be specified in various ways.

ExpO fidelity and stability Plumb et al. [129] proposed two metrics for measuring the desired properties of explanations and using them as regularization terms, to improve the explainability of trained models. These two metrics can also be used as trustworthiness metrics for LIME and its variants, as they are able to evaluate the related fidelity and stability of proxy models. We use ExpO-Fidelity and ExpO-Stability to refer the two metrics in this paragraph, where ExpO is short for Explanation-based Optimization, in order to avoid the confusion to the Infidelity and Sensitivity [183]. The formulas of ExpO-Fidelity and ExpO-Stability are

$$F(\boldsymbol{f}, \boldsymbol{g}, N_x) = \mathbb{E}_{x' \sim N_x}[(\boldsymbol{g}(x') - \boldsymbol{f}(x'))^2], \tag{19}$$

$$F(\boldsymbol{f}, \boldsymbol{e}, N_x) = \mathbb{E}_{x' \sim N_x}[||\boldsymbol{e}(x, \boldsymbol{f}) - \boldsymbol{e}(x', \boldsymbol{f})||_2^2]\|, \tag{20}$$

where $\boldsymbol{g}$ is the proxy model obtained by LIME or its variants, and $\boldsymbol{e}(x, \boldsymbol{f})$ represents the post-hoc local explanation result given a local data point x to explain and the model $\boldsymbol{f}$.

Sensitivity to hyperparameters Besides evaluations on the trustworthiness to the model, Bansal et al. [22] proposed to measure the sensitivity to hyperparameters. "It is important to carefully evaluate the pros and cons of interpretability methods with no hyperparameters and those that have". In fact, the insensitivity to hyperparameters is also an important metric to trustworthiness.

4.2 Model interpretability evaluation

In some situations, different deep models exhibit different abilities to expose understandable terms to humans. Even the same network architecture, training on different datasets may have different interpretability scores [24]. Given the same trustworthy interpretation algorithm and any two models, model interpretability evaluation methods are used to measure and compare the interpretability between models. In this subsection, we introduce four model interpretability evaluation methods, i.e., Network Dissection [24], Pointing Game [189], Consensus [104] and the one through OOD Samples [59, 60].

The basic idea for evaluating the model interpretability for Network Dissection [24], Pointing Game [189] and Consensus [104] is to measure the overlap between semantic items (e.g., segmentation ground truth by humans, or cross-model ensemble of explanations) and interpretation results, as shown in Fig. 3.

Network dissection Network Dissection [24], based on CAM [197], relies on a densely-labeled dataset where each image is labeled across colors, materials, textures, scenes, objects and object parts. Given a CNN model, Network Dissection recovers the intermediate-layer feature maps used by the model for the classification, and then measures the mean intersection over union (mIoU) of each neuron between the activated locations with the labeled visual concepts. A neuron is semantic if its mIoU is larger than a threshold. Then the number of semantic neurons and the ratio are considered as the score for model interpretability.

Pointing game The Pointing Game [189] measures the model interpretability via the localization accuracy. This accuracy is equally the true positive rate between the computed explanation and the annotated object of interest. It is similar to Network Dissection in the way that the pixel-wise or box-wise labels for visual concepts are required and the same intersection between explanations and annotations is measured.

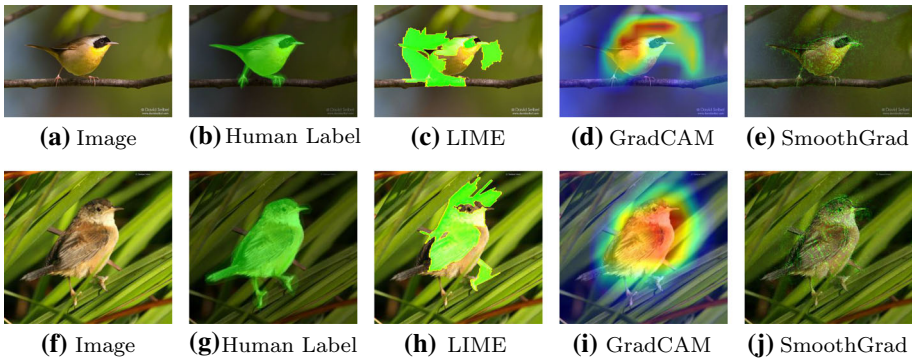


Fig. 3 Visualizations of semantic segmentation ground truth and interpretations from three popular algorithms, i.e., LIME, GradCAM and SmoothGrad, where the interpretation results are shown in different levels of granularity, i.e., superpixel, low-resolution, and pixel, respectively. We use the three algorithms to interpret images from CUB-200-2011 [175], where the semantic segmentations are available

Consensus Consensus approach [104] incorporates an ensemble of deep models as a committee. Consensus first computes interpretations using a trustworthy interpretation algorithm (e.g., LIME [137], SmoothGrad [155]) for every model in the committee, then obtains the consensus of interpretation from the entire committee through voting. Further, Consensus evaluates the interpretability of a model through matching its interpretation result (of LIME or SmoothGrad) to the consensus, and ranks the matching scores together with other deep models in the committee, so as to pursue the absolute and relative interpretability evaluation results. Consensus uses LIME and SmoothGrad to validate its effectiveness, while Consensus is also compatible with other algorithms that interpret other targets, such as the rationale process, as long as the voting approach is suitable for the interpretation algorithm.

Through OOD samples BAM [181] and Trojaning attacks [37, 68] create datasets that are different from natural distributions, and train the models on such datasets. Models trained on such datasets are used to verify the trustworthiness of interpretation algorithms because they should suffer from the attacks on the datasets. In another way, one can use the such ideas of out-of-distribution (OOD) samples to directly evaluate the deep models where the OOD samples were not seen during training. [59, 60] generated different OOD datasets and tested with classic deep models and human observers to record the errors that they made on these datasets. With sophisticated designs of datasets and experiments, they found that the consistency between humans and deep models is closing. These evaluations show the interpretability of deep models in a general way, to present that the visual recognition of models is partially consistent with humans. This could be easily extended to the comparison within models.

4.3 Human-centered/user-study evaluations

User studies involving humans are a commonly used method for evaluating the trustworthiness of interpretation algorithms and model interpretability. We combine these two directions and introduce them here, as the designed user-study experiments may be capable of performing the two evaluations simultaneously.

An approach to evaluate the algorithm of counterfactual examples [11] was proposed, where a user-study experiment was used to validate their approaches. This user-study exper-

Table 3 List of evaluation methods

Method name	Category	Additional notes
Perturbation	T.E.I.A	AUC scores of MoRF, LeRF
Randoming parameters	T.E.I.A	Filtering Irrelavant Algorithms
BAM	T.E.I.A	Based on a semi-natural dataset
Trojaning	T.E.I.A	Based on a semi-natural dataset
Infidelity and sensitivity	T.E.I.A	–
Expo fidelity and stability	T.E.I.A	Available only for LIME and variants
Sensitivity to hyperparameters	T.E.I.A	–
Network dissection	M.I.E	Based on a densely labeled dataset
Pointing game	M.I.E	Requires pixel-wise or box-wise labels
Consensus	M.I.E	Based on cross-model explanations
Through OOD samples	M.I.E	Based on OOD datasets

There are two categories of evaluations as introduced in this work: Trustworthiness Evaluations of Interpretation Algorithm (T.E.I.A) and Model Interpretability Evaluation (M.I.E), with respect to Sect. 4.1 and Sect. 4.2. Additional notes are added as a description for the speciality of the evaluation method

iment aims at verifying whether humans can predict the deep model's decision. Specifically, several (clean and counterfactual) samples with models' predictions are presented to users, and then a new sample is shown to ask the user if the model can make the correct decisions or not. Another approach based on decision trees and sets, designs descriptive and multiple-choice questions to test the user's understanding of the decision boundaries of the classes in the data, in order to evaluate the interpretability of their proposed Bayesian Decision Lists. [56] designed the user-study experiments following the idea that interpretability is the user's ability to predict the model's changes in response to changes in input. More user studies can be found in [66, 78, 94].

4.4 Concluding remarks

We summarize the evaluation methods in Table 3. We have to note that assessing the trustworthiness of interpretation algorithms is challenging. While a small number of algorithms benefit from intrinsic properties of deep models, e.g., closed-form interpretations, the trustworthiness of most algorithms remains to be evaluated. Despite filtering approaches (such as randomizing the weights [3]) to picking out irrelevant interpretation algorithms, reasonable and practical evaluation approaches for directly assessing the trustworthiness are also reviewed. Given a trustworthy algorithm, the interpretability can be evaluated between models, to compare the degree of being understandable. If the algorithm is not trustworthy, it does not make sense to compare the interpretability of models using unreliable interpretation results. A few model interpretability evaluation methods are introduced, while more model interpretability evaluations should be explored in the future. We also note that subjective human-centered user studies are one important evaluation tool that can be used for evaluating both interpretation algorithms and model interpretability, thanks to the flexibility of designing arbitrary experiments for various objectives.

5 Impact beyond interpretations

Deep models have many unknown phenomenon and properties, e.g., adversarial attacks, memorization capacity, generalization ability etc. (Lack of) interpretation and (low) interpretability are one of them. Interestingly, besides the original motivations for explaining black-box deep models, interpretation-related terms have been connected to existing findings about deep models. In this section, we present two fields that are widely known to be related to interpretations.

5.1 Interpretability, adversarial attacks, and robustness

Recent studies on adversarial examples have found positive connections between model interpretability and adversarial robustness. Two teams [140, 165] first observed that compared to standard models, adversarially trained models show more interpretable input gradients. Etmann et al. [50] theoretically proved that the increase in adversarial robustness improves the alignment between input and its respective input gradient, using the case of a linear binary classifier. Zhang et al. [194] further analyzed how adversarially trained models achieve robustness from an interpretation perspective, showing that adversarially robust models rely on fewer texture features and are more shape-biased, which is regarded as coincide more with the human interpretation. Essentially, the connection between adversarial examples and gradient-based interpretations may come from their common dependence on the input gradient.

For future works, these observations could (1) motivate new understandings about how deep models work and (2) explore the connections between interpretation-related terms and other properties of deep models.

5.2 Learning from interpretations

As containing rich information about the location of discriminative features, interpretation results can also be utilized to guide training strategies such as data augmentations and regularization approaches, especially for vision tasks. For example, Kim et al. [89] proposed to improve Mixup [188] by leveraging the saliency map [153]. Specifically, they aimed to seek an optimal transport that maximizes the exposed saliency. Zagoruyko et al. [186] imposed the regularizer to encourage the alignment of saliency maps between the teacher and student networks for effective knowledge distillation. Wickramanayake et al. [176] also used interpretations to generate efficient augmented data samples to train the model, for improving the interpretability and the model performance. Interpretations sometimes can be used as weak labels in specific tasks. For example, Lai et al. [95] introduced a saliency-guided learning approach for weakly supervised object detection. Many weakly object localization and weakly semantic segmentation methods [7, 89, 182] start from an interpretation, and obtain promising results.

From these works, we believe that the interpretability and model performance are not two contradictory measures and that they can be improved simultaneously. Future works could further focus on this direction.

6 Open-source libraries for deep learning interpretation

To simplify future researches and practical usages, we introduce several open-source libraries that implement popular interpretation algorithms based on mainstream deep learning frameworks, such as TF-Explainer[4] based on Tensorflow [1], Captum[5] based on PyTorch [125] and InterpretDL[6] based on PaddlePaddle [112]. Note that TF-explainer and Captum mainly include algorithms that target at features with gradient-based techniques. Some other popular libraries focus on machine learning and have not involved deep models, such as interpretml,[7] AIX360[8] etc., and the library LIT[9] that is for NLP models.

7 Discussions and conclusions

In this paper, we review the recent research on interpretation algorithms, model interpretability, and the connections to other deep learning factors.

First of all, to address the research efforts in interpretations, we clarify the main concepts of interpretation algorithms and model interpretability that were usually confused, and connect them by introducing the notion of trustworthiness of interpretation algorithms.

Second, we propose a new taxonomy and elaborate the design of several recent interpretation algorithms, from different perspectives according to the proposed taxonomy. Our work reviews the recent advances in interpretation algorithms, and provides a clear categorization, to help future researches to better compare new algorithms with the most related works, or progress in unexplored directions.

Third, we survey the performance metrics for evaluating the trustworthiness of interpretation algorithms, to guarantee the appropriate usages of the interpretation results. These metrics can be used to quantitatively compare between the interpretation algorithms. The proposition of new algorithms can be supported by comparing these metrics with related works, instead of by providing tenuous descriptions and qualitative visualizations.

Further, we summarize the current work in evaluating models' interpretability given trustworthy interpretation algorithms. Based on these evaluations, more relations between interpretability and other metrics could be found for deep models, possibly leading to further understandings about the deep learning. However, there are not many evaluation methods for measuring the interpretability, though the existing ones are largely aligned for popular network architectures. Designing new methods of evaluating models' interpretability could be one of the important research directions.

Finally, we review and discuss the connections between deep models' interpretations and other factors, such as adversarial robustness and learning from interpretations. New understandings how deep models could be observed and analyzed. Note that many interpretation algorithms and evaluation approaches are open-sourced and there are some useful libraries to simplify the practical usages and future researches.

Acknowledgments Funding was provided by National Key R&D Program of China (Grant No. 2021ZD0110303).

[4] https://github.com/sicara/tf-explain.

[5] https://github.com/pytorch/captum.

[6] https://github.com/PaddlePaddle/InterpretDL.

[7] https://github.com/interpretml/interpret.

[8] https://github.com/Trusted-AI/AIX360.

[9] https://github.com/PAIR-code/lit.

References

1. Abadi M, Agarwal A, Barham P, Brevdo E, Chen Z, Citro C, Corrado GS, Davis A, Dean J, Devin M, Ghemawat S, Goodfellow I, Harp A, Irving G, Isard M, Jia Y, Jozefowicz R, Kaiser L, Kudlur M, Levenberg J, Mané D, Monga R, Moore S, Murray D, Olah C, Schuster M, Shlens J, Steiner B, Sutskever I, Talwar K, Tucker P, Vanhoucke V, Vasudevan V, Viégas F, Vinyals O, Warden P, Wattenberg M, Wicke M, Yu Y, Zheng X (2015) TensorFlow: large-scale machine learning on heterogeneous systems. Software available from tensorflow.org
2. Abnar S, Zuidema WH (2020) Quantifying attention flow in transformers. In: Jurafsky D, Chai J, Schluter N, Tetreault JR (eds) Proceedings of the 58th annual meeting of the association for computational linguistics, ACL. Association for Computational Linguistics
3. Adebayo J, Gilmer J, Muelly M, Goodfellow IJ, Hardt M, Kim B (2018) Sanity checks for saliency maps. In: Bengio S, Wallach HM, Larochelle H, Grauman K, Cesa-Bianchi N, Garnett R (eds) Advances in neural information processing systems 31: annual conference on neural information processing systems 2018 (NeurIPS 2018), December 3–8, 2018, Montréal, Canada, pp 9525–9536 (2018)
4. Adebayo J, Muelly M, Liccardi I, Kim B (2020) Debugging tests for model explanations. In: Larochelle H, Ranzato M, Hadsell R, Balcan M-F, Lin H-T (eds) Advances in neural information processing systems 33: annual conference on neural information processing systems 2020 (NeurIPS 2020), December 6–12, 2020
5. Agarwal R, Melnick L, Frosst N, Zhang X, Lengerich BJ, Caruana R, Hinton GE (2021) Neural additive models: interpretable machine learning with neural nets. In: Ranzato M, Beygelzimer A, Dauphin YN, Liang P, Vaughan JW (eds) Advances in neural information processing systems 34: annual conference on neural information processing systems 2021 (NeurIPS 2021), December 6–14, 2021, pp 4699–4711 (2021)
6. Ahern I, Noack A, Guzman-Nateras L, Dou D, Li B, Huan J (2019) Normlime: a new feature importance metric for explaining deep neural networks. CoRR, arXiv:1909.04200
7. Ahn J, Kwak S (2018) Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation. In: 2018 IEEE conference on computer vision and pattern recognition (CVPR 2018), Salt Lake City, UT, USA, June 18–22, 2018. Computer Vision Foundation/IEEE Computer Society, pp 4981–4990
8. Alvarez-Melis D, Jaakkola TS (2018) On the robustness of interpretability methods. CoRR arXiv:1806.08049
9. Ancona M, Ceolini E, Öztireli C, Gross M (2018) Towards better understanding of gradient-based attribution methods for deep neural networks. In: 6th International conference on learning representations (ICLR 2018), Vancouver, BC, Canada, April 30–May 3, 2018, conference track proceedings. OpenReview.net
10. Andrychowicz M, Baker B, Chociej M, Józefowicz R, McGrew B, Pachocki J, Petron A, Plappert M, Powell G, Ray A, Schneider J, Sidor S, Tobin J, Welinder P, Weng L, Zaremba W (2020) Learning dexterous in-hand manipulation. Int J Robot Res 39(1):66
11. Antorán J, Bhatt U, Adel T, Weller A, Hernández-Lobato JM (2021) Getting a CLUE: a method for explaining uncertainty estimates. In: 9th International conference on learning representations (ICLR 2021), virtual event, Austria, May 3–7, 2021. OpenReview.net
12. André A, Barbara H (2019) On the computation of counterfactual explanations—a survey. CoRR, arXiv:1911.07749 (2019)
13. Arulkumaran K, Deisenroth MP, Brundage M, Bharath AA (2017) Deep reinforcement learning: a brief survey. IEEE Signal Process Mag 34(6):26–38
14. Atanasova P, Simonsen JG, Lioma C, Augenstein I (2020) Generating fact checking explanations. In: Jurafsky D, Chai J, Schluter N, Tetreault JR (eds) Proceedings of the 58th annual meeting of the association for computational linguistics (ACL). Association for Computational Linguistics
15. Atrey A, Clary K, Jensen DD (2020) Exploratory not explanatory: counterfactual analysis of saliency maps for deep reinforcement learning. In: 8th International conference on learning representations (ICLR 2020), Addis Ababa, Ethiopia, April 26–30, 2020. OpenReview.net
16. Bach S, Binder A, Montavon G, Klauschen F, Müller K-R, Samek W (2015) On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. PLoS ONE 6:66
17. Bahdanau D, Cho K, Bengio Y (2015) Neural machine translation by jointly learning to align and translate. In: Bengio Y, LeCun Y (eds) International conference on learning representations
18. Bajaj M, Chu L, Xue ZY, Pei J, Wang L, Lam PC-H, Zhang Y (2021) Robust counterfactual explanations on graph neural networks. In: Ranzato MA, Beygelzimer A, Dauphin YN, Liang P, Vaughan JW (eds) Advances in neural information processing systems 34: annual conference on neural information processing systems 2021 (NeurIPS 2021), December 6–14, 2021, virtual, pp 5644–5655

19. Balabanović M, Shoham Y (1997) Fab: content-based, collaborative recommendation. Commun ACM 6:66
20. Baldassarre F, Azizpour H (2019) Explainability techniques for graph convolutional networks. CoRR, arXiv:1905.13686
21. Baldock RJN, Maennel H, Neyshabur B (2021) Deep learning through the lens of example difficulty. In: Ranzato MA, Beygelzimer A, Dauphin YN, Liang P, Vaughan JW (eds) Advances in neural information processing systems 34: annual conference on neural information processing systems 2021 (NeurIPS 2021), December 6–14, 2021, virtual, pp 10876–10889
22. Bansal N, Agarwal C, Nguyen A (2020) SAM: the sensitivity of attribution methods to hyperparameters. In: 2020 IEEE/CVF conference on computer vision and pattern recognition (CVPR 2020), Seattle, WA, USA, June 13–19, 2020, pp 8670–8680. Computer Vision Foundation/IEEE
23. Barbalau A, Cosma A, Ionescu RT, Popescu M (2020) A generic and model-agnostic exemplar synthetization framework for explainable AI. In: Hutter F, Kersting K, Lijffijt J, Valera I (eds) Machine learning and knowledge discovery in databases—European conference, ECML PKDD 2020, Ghent, Belgium, September 14–18, 2020, Proceedings, Part II, volume 12458 of lecture notes in computer science. Springer, pp 190–205
24. Bau D, Zhou B, Khosla A, Oliva A, Torralba A (2017) Network dissection: quantifying interpretability of deep visual representations. In: 2017 IEEE conference on computer vision and pattern recognition (CVPR 2017), Honolulu, HI, USA, July 21–26, 2017. IEEE Computer Society, pp 3319–3327
25. Bau D, Zhu J-Y, Strobelt H, Zhou B, Tenenbaum JB, Freeman WT, Torralba A (2019) GAN dissection: visualizing and understanding generative adversarial networks. In: 7th International conference on learning representations (ICLR 2019), New Orleans, LA, USA, May 6–9, 2019. OpenReview.net (2019)
26. Bien J, Tibshirani R (2011) Prototype selection for interpretable classification. Ann Appl Stat 6:66
27. Binder A, Montavon G, Lapuschkin S, Müller K-R, Samek W (2016) Layer-wise relevance propagation for neural networks with local renormalization layers. In: Villa AEP, Masulli P, Rivero AP (eds) Artificial neural networks and machine learning—ICANN 2016—25th international conference on artificial neural networks, Barcelona, Spain, September 6–9, 2016, Proceedings, Part II, volume 9887 of lecture notes in computer science. Springer, pp 63–71
28. Carlini N, Erlingsson Ú, Papernot N (2019) Distribution density, tails, and outliers in machine learning: metrics and applications. CoRR, arXiv:1910.13427
29. Carvalho DV, Pereira EM, Cardoso JS (2019) Machine learning interpretability: a survey on methods and metrics. Electronics 6:66
30. Chakraborty A, Alam M, Dey V, Chattopadhyay A, Mukhopadhyay D (2018) Adversarial attacks and defences: a survey. CoRR, arXiv:1810.00069
31. Chang C-H, Creager E, Goldenberg A, Duvenaud D (2019) Explaining image classifiers by counterfactual generation. In: 7th International conference on learning representations (ICLR 2019), New Orleans, LA, USA, May 6–9, 2019. OpenReview.net
32. Chattopadhyay A, Sarkar A, Howlader P, Balasubramanian VN (2018) Grad-CAM++: generalized gradient-based visual explanations for deep convolutional networks. In: 2018 IEEE winter conference on applications of computer vision (WACV 2018), Lake Tahoe, NV, USA, March 12–15, 2018. IEEE Computer Society, pp 839–847
33. Chefer H, Gur S, Wolf L (2021) Generic attention-model explainability for interpreting bi-modal and encoder-decoder transformers. In: 2021 IEEE/CVF international conference on computer vision (ICCV 2021), Montreal, QC, Canada, October 10–17, 2021. IEEE, pp 387–396
34. Chefer H, Gur S, Wolf L (2021) Transformer interpretability beyond attention visualization. In: IEEE conference on computer vision and pattern recognition (CVPR 2021), virtual, June 19–25, 2021. Computer Vision Foundation/IEEE, pp 782–791
35. Chen C, Li O, Tao D, Barnett A, Rudin C, Su J (2019) This looks like that: deep learning for interpretable image recognition. In: Wallach HM, Larochelle H, Beygelzimer A, d'Alché-Buc F, Fox EB, Garnett R (eds) Advances in neural information processing systems 32: annual conference on neural information processing systems 2019 (NeurIPS 2019), December 8–14, 2019, Vancouver, BC, Canada, pp 8928–8939
36. Chen C, Zhang M, Liu Y, Ma S (2018) Neural attentional rating regression with review-level explanations. In: Champin P-A, Gandon F, Lalmas M, Ipeirotis PG (eds) Proceedings of the 2018 World Wide Web conference on World Wide Web (WWW 2018), Lyon, France, April 23–27, 2018. ACM, pp 1583–1592
37. Chen X, Liu C, Li B, Lu K, Song D (2017) Targeted backdoor attacks on deep learning systems using data poisoning. CoRR, arXiv:1712.05526
38. Chen Y, Li B, Yu H, Wu P, Miao C (2021) Hydra: hypergradient data relevance analysis for interpreting deep neural networks. In: Thirty-fifth AAAI conference on artificial intelligence (AAAI 2021), thirty-third conference on innovative applications of artificial intelligence (IAAI 2021), the eleventh symposium

on educational advances in artificial intelligence (EAAI 2021), virtual event, February 2–9, 2021. AAAI Press, pp 7081–7089 (2021)
39. Cheng H-T, Koc L, Harmsen J, Shaked T, Chandra T, Aradhye H, Anderson G, Corrado G, Chai W, Ispir M, Anil R, Haque Z, Hong L, Jain V, Liu X, Shah H (2016) Wide & deep learning for recommender systems. In: Karatzoglou A, Hidasi B, Tikk D, Shalom OS, Roitman H, Shapira B, Rokach L (eds) Proceedings of the 1st workshop on deep learning for recommender systems, DLRS@RecSys 2016, Boston, MA, USA, September 15, 2016. ACM, pp 7–10
40. Covington P, Adams J, Sargin E (2016) Deep neural networks for youtube recommendations. In: Sen S, Geyer W, Freyne J, Castells P (eds) Proceedings of the 10th ACM conference on recommender systems, Boston, MA, USA, September 15–19, 2016. ACM, pp 191–198
41. Croce F, Andriushchenko M, Sehwag V, Debenedetti E, Flammarion N, Chiang M, Mittal P, Hein M (2020) Robustbench: a standardized adversarial robustness benchmark. arXiv preprint arXiv:2010.09670
42. Dacrema MF, Cremonesi P, Jannach D (2019) Are we really making much progress? A worrying analysis of recent neural recommendation approaches. In: Bogers T, Said A, Brusilovsky P, Tikk D (eds) Proceedings of the 13th ACM conference on recommender systems (RecSys 2019), Copenhagen, Denmark, September 16–20, 2019. ACM, pp 101–109
43. Deng J, Dong W, Socher R, Li L-J, Li K, Fei-Fei L (2009) Imagenet: a large-scale hierarchical image database. In: 2009 IEEE computer society conference on computer vision and pattern recognition (CVPR 2009), 20–25 June 2009, Miami, Florida, USA. IEEE Computer Society, pp 248–255
44. Desai S, Ramaswamy HG (2020) Ablation-CAM: visual explanations for deep convolutional network via gradient-free localization. In: IEEE winter conference on applications of computer vision (WACV 2020), Snowmass Village, CO, USA, March 1–5, 2020. IEEE, pp 972–980 (2020)
45. Devlin J, Chang M-W, Lee K, Toutanova K (2019) BERT: pre-training of deep bidirectional transformers for language understanding. In: Burstein J, Doran C, Solorio T (eds) Proceedings of the 2019 conference of the North American Chapter of the Association for computational linguistics: human language technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2–7, 2019, Volume 1 (Long and Short Papers). Association for Computational Linguistics, pp 4171–4186
46. Dong Y, Su H, Zhu J, Zhang B (2017) Improving interpretability of deep neural networks with semantic information. In: 2017 IEEE conference on computer vision and pattern recognition (CVPR 2017), Honolulu, HI, USA, July 21–26, 2017. IEEE Computer Society, pp 975–983
47. Doshi-Velez F, Kim B (2017) Towards a rigorous science of interpretable machine learning. arXiv preprint arXiv:1702.08608
48. Dosovitskiy A, Beyer L, Kolesnikov A, Weissenborn D, Zhai X, Unterthiner T, Dehghani M, Minderer M, Heigold G, Gelly S, Uszkoreit J, Houlsby N (2021) An image is worth 16x16 words: transformers for image recognition at scale. In: 9th International conference on learning representations (ICLR 2021), virtual event, Austria, May 3–7, 2021. OpenReview.net
49. Erhan D, Bengio Y, Courville A, Vincent P (2009) Visualizing higher-layer features of a deep network. In: 2018 IEEE international conference on machine learning workshops
50. Etmann C, Lunz S, Maass P, Schönlieb C (2019) On the connection between adversarial robustness and saliency map interpretability. In: Chaudhuri K, Salakhutdinov R (eds) Proceedings of the 36th international conference on machine learning (ICML 2019), 9–15 June 2019, Long Beach, CA, USA, volume 97 of proceedings of machine learning research (PMLR), pp 1823–1832
51. Faber L, Moghaddam AK, Wattenhofer R (2020) Contrastive graph neural network explanation. CoRR, arXiv:2010.13663
52. Feldman V (2020) Does learning require memorization? A short tale about a long tail. In: Makarychev K, Makarychev Y, Tulsiani M, Kamath G, Chuzhoy J (eds) Proceedings of the 52nd annual ACM SIGACT symposium on theory of computing (STOC 2020), Chicago, IL, USA, June 22–26, 2020. ACM, pp 954–959
53. Feldman V, Zhang C (2020) What neural networks memorize and why: discovering the long tail via influence estimation. In: Larochelle H, Ranzato MA, Hadsell R, Balcan M-F, Lin H-T (eds) Advances in neural information processing systems 33: annual conference on neural information processing systems 2020 (NeurIPS 2020), December 6–12, 2020, virtual
54. Fong R, Patrick M, Vedaldi A (2019) Understanding deep networks via extremal perturbations and smooth masks. In: 2019 IEEE/CVF international conference on computer vision (ICCV 2019), Seoul, Korea (South), October 27–November 2, 2019. IEEE, pp 2950–2958
55. Fong RC, Vedaldi A (2017) Interpretable explanations of black boxes by meaningful perturbation. In: IEEE international conference on computer vision (ICCV 2017), Venice, Italy, October 22–29, 2017. IEEE Computer Society, pp 3449–3457
56. Friedler SA, Roy CD, Scheidegger C, Slack D (2019) Assessing the local interpretability of machine learning models. CoRR, arXiv:1902.03501

57. Frosst N, Hinton GE (2017) Distilling a neural network into a soft decision tree. In: Besold TR, Kutz O (eds) Proceedings of the first international workshop on comprehensibility and explanation in AI and ML 2017 co-located with 16th international conference of the Italian Association for artificial intelligence (AI*IA 2017), Bari, Italy, November 16th and 17th, 2017, volume 2071 of CEUR workshop proceedings. CEUR-WS.org
58. Geirhos R, Jacobsen J-H, Michaelis C, Zemel RS, Brendel W, Bethge M, Wichmann FA (2020) Shortcut learning in deep neural networks. Nat Mach Intell 2(11):665–673
59. Geirhos R, Narayanappa K, Mitzkus B, Thieringer T, Bethge M, Wichmann FA, Brendel W (2021) Partial success in closing the gap between human and machine vision. In: Ranzato MA, Beygelzimer A, Dauphin YN, Liang P, Vaughan JW (eds) Advances in neural information processing systems 34: annual conference on neural information processing systems 2021 (NeurIPS 2021), December 6–14, 2021, virtual, pp 23885–23899
60. Geirhos R, Rubisch P, Michaelis C, Bethge M, Wichmann FA, Brendel W (2019) Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. In: 7th International conference on learning representations (ICLR 2019), New Orleans, LA, USA, May 6–9, 2019. OpenReview.net
61. Ghaeini R, Fern XZ, Tadepalli P (2018) Interpreting recurrent and attention-based neural models: a case study on natural language inference. In: Riloff E, Chiang D, Hockenmaier J, Tsujii J (eds) Proceedings of the 2018 conference on empirical methods in natural language processing. Association for Computational Linguistics
62. Goh GB, Hodas NO, Vishnu A (2017) Deep learning for computational chemistry. J Comput Chem 38(16):1291–1307
63. Gomez-Uribe CA, Hunt N (2016) The netflix recommender system: algorithms, business value, and innovation. ACM Trans Manag Inf Syst 6(4):131–1319
64. Goyal Y, Wu Z, Ernst J, Batra D, Parikh D, Lee S (2019) Counterfactual visual explanations. In: Chaudhuri K, Salakhutdinov R (eds) Proceedings of the 36th international conference on machine learning (ICML 2019), 9–15 June 2019, Long Beach, CA, USA, volume 97 of proceedings of machine learning research (PMLR), pp 2376–2384
65. Greydanus S, Koul A, Dodge J, Fern A (2018) Visualizing and understanding Atari agents. In: Dy JG, Krause A (eds) Proceedings of the 35th international conference on machine learning (ICML 2018), Stockholmsmässan, Stockholm, Sweden, July 10–15, 2018, volume 80 of proceedings of machine learning research (PMLR), pp 1787–1796
66. Grgic-Hlaca N, Redmiles EM, Gummadi KP, Weller A (2018) Human perceptions of fairness in algorithmic decision making: a case study of criminal risk prediction. In: Champin P-A, Gandon F, Lalmas M, Ipeirotis PG (eds) Proceedings of the 2018 World Wide Web conference on World Wide Web (WWW 2018), Lyon, France, April 23–27, 2018. ACM, pp 903–912
67. Gu J, Yang Y, Tresp V (2018) Understanding individual decisions of cnns via contrastive backpropagation. In: Jawahar CV, Li H, Mori G, Schindler K (eds) Computer vision—ACCV 2018—14th Asian conference on computer vision, Perth, Australia, December 2–6, 2018, revised selected papers, Part III, volume 11363 of lecture notes in computer science. Springer, pp 119–134
68. Gu T, Dolan-Gavitt B, Garg S (2017) BadNets: identifying vulnerabilities in the machine learning model supply chain. CoRR, arXiv:1708.06733
69. Guidotti R, Monreale A, Matwin S, Pedreschi D (2019) Black box explanation by learning image exemplars in the latent feature space. In: Brefeld U, Fromont É, Hotho A, Knobbe AJ, Maathuis MH, Robardet C (eds) Machine learning and knowledge discovery in databases—European conference (ECML PKDD 2019), Würzburg, Germany, September 16–20, 2019, proceedings, Part I, volume 11906 of lecture notes in computer science. Springer, pp 189–205
70. Hendrycks D, Dietterich TG (2019) Benchmarking neural network robustness to common corruptions and perturbations. In: 7th International conference on learning representations (ICLR 2019), New Orleans, LA, USA, May 6–9, 2019. OpenReview.net
71. Heo J, Joo S, Moon T (2019) Fooling neural network interpretations via adversarial model manipulation. In: Wallach HM, Larochelle H, Beygelzimer A, d'Alché-Buc F, Fox EB, Garnett R (eds) Advances in neural information processing systems 32: annual conference on neural information processing systems 2019 (NeurIPS 2019), December 8–14, 2019, Vancouver, BC, Canada, pp 2921–2932
72. Herlocker JL, Konstan JA, Riedl J (2000) Explaining collaborative filtering recommendations. In: Kellogg WA, Whittaker S (eds) CSCW 2000, proceeding on the ACM 2000 conference on computer supported cooperative work, Philadelphia, PA, USA, December 2–6, 2000. ACM, pp 241–250 (2000)
73. Hinton GE, Sabour S, Frosst N (2018) Matrix capsules with EM routing. In: 6th International conference on learning representations (ICLR 2018), Vancouver, BC, Canada, April 30–May 3, 2018, conference track proceedings. OpenReview.net

74. Hooker S, Erhan D, Kindermans P-J, Kim B (2019) A benchmark for interpretability methods in deep neural networks. In: Wallach HM, Larochelle H, Beygelzimer A, d'Alché-Buc F, Fox EB, Garnett R (eds) Advances in neural information processing systems 32: annual conference on neural information processing systems 2019 (NeurIPS 2019), December 8–14, 2019, Vancouver, BC, Canada, pp 9734–9745
75. Hua K-L, Hsu C-H, Hidayati SC, Wen-Huang C, Yu-Jen C (2015) Computer-aided classification of lung nodules on computed tomography images via deep learning technique. OncoTargets Ther 6:66
76. Huang Q, Yamada M, Tian Y, Singh D, Yin D, Chang Y (2020) Graphlime: local interpretable model explanations for graph neural networks. CoRR, arXiv:2001.06216
77. Ilyas A, Santurkar S, Tsipras D, Engstrom L, Tran B, Madry A (2019) Adversarial examples are not bugs, they are features. In: Wallach HM, Larochelle H, Beygelzimer A, d'Alché-Buc F, Fox EB, Garnett R (eds) Advances in neural information processing systems 32: annual conference on neural information processing systems 2019 (NeurIPS 2019), December 8–14, 2019, Vancouver, BC, Canada, pp 125–136
78. Islam SR, Eberle W, Ghafoor SK (2020) Towards quantification of explainability in explainable artificial intelligence methods. In: Barták R, Bell E (eds) Proceedings of the thirty-third international Florida artificial intelligence research society conference, originally to be held in North Miami Beach, Florida, USA, May 17–20, 2020. AAAI Press, pp 75–81
79. Iwana BK, Kuroki R, Uchida S (2019) Explaining convolutional neural networks using softmax gradient layer-wise relevance propagation. In: 2019 IEEE/CVF international conference on computer vision workshops (ICCV Workshops 2019), Seoul, Korea (South), October 27–28, 2019. IEEE, pp 4176–4185
80. Iyer R, Li Y, Li H, Lewis M, Sundar R, Sycara KP (2018) Transparency and explanation in deep reinforcement learning neural networks. In: Furman J, Marchant GE, Price H, Rossi F (eds) Proceedings of the 2018 AAAI/ACM conference on AI, ethics, and society, AIES 2018, New Orleans, LA, USA, February 02–03, 2018. ACM, pp 144–150
81. Jacovi A, Goldberg Y (2020) Towards faithfully interpretable NLP systems: How should we define and evaluate faithfulness? In: Jurafsky D, Chai J, Schluter N, Tetreault JR (eds) Proceedings of the 58th annual meeting of the association for computational linguistics (ACL 2020), Online, July 5–10, 2020. Association for Computational Linguistics, pp 4198–4205
82. Jain S, Wallace BC (2019) Attention is not explanation. In: Burstein J, Doran C, Solorio T (eds) Proceedings of the 2019 conference of the North American Chapter of the Association for computational linguistics: human language technologies, NAACL-HLT. Association for Computational Linguistics
83. Jo T, Nho K, Saykin AJ (2019) Deep learning in Alzheimer's disease: diagnostic classification and prognostic prediction using neuroimaging data. CoRR, arXiv:1905.00931
84. Jumper J, Evans R, Pritzel A, Green T, Figurnov M, Ronneberger O, Tunyasuvunakool K, Bates R, Žídek A, Potapenko A et al (2021) Highly accurate protein structure prediction with alphafold. Nature 6:66
85. Kaelbling LP, Littman ML, Moore AW (1996) Reinforcement learning: a survey. J Artif Intell Res 4:237–285
86. Khan A, Huerta EA, Zheng H (2021) Interpretable AI forecasting for numerical relativity waveforms of quasi-circular, spinning, non-precessing binary black hole mergers. CoRR, arXiv:2110.06968
87. Kim B, Wattenberg M, Gilmer J, Cai CJ, Wexler J, Viégas FB, Sayres R (2018) Interpretability beyond feature attribution: quantitative testing with concept activation vectors (TCAV). In: Dy JG, Krause A (eds) Proceedings of the 35th international conference on machine learning (ICML 2018), Stockholmsmässan, Stockholm, Sweden, July 10–15, 2018, volume 80 of Proceedings of machine learning research, pp 2673–2682
88. Kim J-M, Choe J, Akata Z, Oh SJ (2021) Keep CALM and improve visual feature attribution. In: 2021 IEEE/CVF international conference on computer vision (ICCV 2021), Montreal, QC, Canada, October 10–17, 2021. IEEE, pp 8330–8340
89. Kim J-H, Choo W, Song HO (2020) Puzzle mix: exploiting saliency and local statistics for optimal mixup. In: Proceedings of the 37th international conference on machine learning (ICML 2020), 13–18 July 2020, Virtual Event, volume 119 of proceedings of machine learning research (PMLR), pp 5275–5285
90. Koh PW, Ang K-S, Teo HHK, Liang P (2019) On the accuracy of influence functions for measuring group effects. In: Wallach HM, Larochelle H, Beygelzimer A, d'Alché-Buc F, Fox EB, Garnett R (eds) Advances in neural information processing systems 32: annual conference on neural information processing systems 2019 (NeurIPS 2019), December 8–14, 2019, Vancouver, BC, Canada, pp 5255–5265
91. Koh PW, Liang P (2017) Understanding black-box predictions via influence functions. In: Precup D, Teh YW (eds) Proceedings of the 34th international conference on machine learning (ICML 2017), Sydney, NSW, Australia, 6–11 August 2017, volume 70 of proceedings of machine learning research (PMLR), pp 1885–1894

92. Kontschieder P, Fiterau M, Criminisi A, Bulò SR (2015) Deep neural decision forests. In: 2015 IEEE international conference on computer vision (ICCV 2015), Santiago, Chile, December 7–13, 2015. IEEE Computer Society, pp 1467–1475
93. Kumar S, Talukdar PP (2020) NILE: natural language inference with faithful natural language explanations. In: Jurafsky D, Chai J, Schluter N, Tetreault JR (eds) Proceedings of the 58th annual meeting of the association for computational linguistics. Association for Computational Linguistics (ACL)
94. Lage I, Chen E, He J, Narayanan M, Kim B, Gershman S, Doshi-Velez F (2019) An evaluation of the human-interpretability of explanation. CoRR, arXiv:1902.00006
95. Lai B, Gong X (2017) Saliency guided end-to-end learning for weakly supervised object detection. In: Sierra C (ed) Proceedings of the twenty-sixth international joint conference on artificial intelligence (IJCAI 2017), Melbourne, Australia, August 19–25, 2017, pp 2053–2059. ijcai.org
96. Lakkaraju H, Kamar E, Caruana R, Leskovec J (2017) Interpretable & explorable approximations of black box models. CoRR, arXiv:1707.01154
97. Laugel T, Lesot M-J, Marsala C, Renard X, Detyniecki M (2019) Unjustified classification regions and counterfactual explanations in machine learning. In: Brefeld U, Fromont É, Hotho A, Knobbe AJ, Maathuis MH, Robardet C (eds) Machine learning and knowledge discovery in databases—European conference (ECML PKDD 2019), Würzburg, Germany, September 16–20, 2019, Proceedings, Part II, volume 11907 of lecture notes in computer science. Springer, pp 37–54
98. LeCun Y, Bengio Y, Hinton GE (2015) Deep learning. Nature 521(7553):436–444
99. Levine S, Finn C, Darrell T, Abbeel P (2016) End-to-end training of deep visuomotor policies. J Mach Learn Res 17:39:1-39:40
100. Levine S, Pastor P, Krizhevsky A, Ibarz J, Quillen D (2018) Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection. Int J Robot Res 37(4–5):421–436
101. Li B, Qi P, Liu B, Di S, Liu J, Pei J, Yi J, Zhou B (2021) Trustworthy AI: from principles to practices. CoRR, arXiv:2110.01167
102. Li C, Quan C, Peng L, Qi Y, Deng Y, Wu L (2019) A capsule network for recommendation and explaining what you like and dislike. In: Piwowarski B, Chevalier M, Gaussier É, Maarek Y, Nie J-Y, Scholer F (eds) Proceedings of the 42nd international ACM SIGIR conference on research and development in information retrieval (SIGIR 2019), Paris, France, July 21–25, 2019. ACM, pp 275–284
103. Li O, Liu H, Chen C, Rudin C (2018) Deep learning for case-based reasoning through prototypes: a neural network that explains its predictions. In: McIlraith SA, Weinberger KQ (eds) Proceedings of the thirty-second AAAI conference on artificial intelligence (AAAI-18), the 30th innovative applications of artificial intelligence (IAAI-18), and the 8th AAAI symposium on educational advances in artificial intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2–7, 2018. AAAI Press, pp 3530–3537
104. Li X, Xiong H, Huang S, Ji S, Dou D (2021) Cross-model consensus of explanations and beyond for image classification models: an empirical study. CoRR, arXiv:2109.00707
105. Li Y (2017) Deep reinforcement learning: an overview. CoRR, arXiv:1701.07274
106. Lin Y-S, Lee W-C, Celik ZB (2021) What do you see? Evaluation of explainable artificial intelligence (XAI) interpretability through neural backdoors. In: Zhu F, Ooi BC, Miao C (eds) KDD '21: the 27th ACM SIGKDD conference on knowledge discovery and data mining, virtual event, Singapore, August 14–18, 2021. ACM, pp 1027–1035
107. Lipton ZC (2018) The mythos of model interpretability. Commun ACM 61(10):36–43
108. Litjens G, Kooi T, Bejnordi BE, Adiyoso Setio AA, Ciompi F, Ghafoorian M, van der Laak JAWM, van Ginneken B, Sánchez CI (2017) A survey on deep learning in medical image analysis. Med Image Anal 42:60–88
109. Liu H, Yin Q, Wang WY (2019) Towards explainable NLP: a generative explanation framework for text classification. In: Korhonen A, Traum DR, Màrquez L (eds) Proceedings of the 57th conference of the association for computational linguistics. Association for Computational Linguistics (ACL)
110. Lundberg SM, Lee S-I (2017) A unified approach to interpreting model predictions. In: Guyon I, von Luxburg U, Bengio S, Wallach HM, Fergus R, Vishwanathan SVN, Garnett R (eds) Advances in neural information processing systems 30: annual conference on neural information processing systems 2017, December 4–9 (2017), Long Beach, CA, USA, pp 4765–4774
111. Luo D, Cheng W, Xu D, Yu W, Zong B, Chen H, Zhang X (2020) Parameterized explainer for graph neural network. In: Larochelle H, Ranzato MA, Hadsell R, Balcan M-F, Lin H-T (eds) Advances in neural information processing systems 33: annual conference on neural information processing systems 2020 (NeurIPS 2020), December 6–12, 2020, virtual
112. Ma Y, Yu D, Wu T, Wang H (2019) Paddlepaddle: an open-source deep learning platform from industrial practice. Front Data Comput 6:66

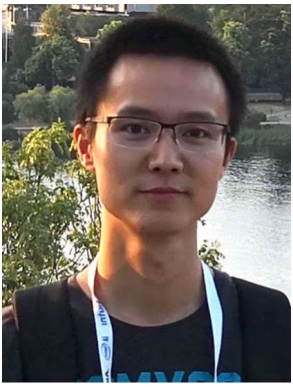
113. Mahendran A, Vedaldi A (2015) Understanding deep image representations by inverting them. In: IEEE conference on computer vision and pattern recognition (CVPR 2015), Boston, MA, USA, June 7–12, 2015. IEEE Computer Society, pp 5188–5196
114. Margeloiu A, Simidjievski N, Jamnik M, Weller A (2020) Improving interpretability in medical imaging diagnosis using adversarial training. CoRR, arXiv:2012.01166
115. Miller T (2019) Explanation in artificial intelligence: insights from the social sciences. Artif Intell 267:1–38
116. Ming Y, Xu P, Qu H, Ren L (2019) Interpretable and steerable sequence learning via prototypes. In: Teredesai A, Kumar V, Li Y, Rosales R, Terzi E, Karypis G (eds) Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining (KDD 2019), Anchorage, AK, USA, August 4–8, 2019. ACM, pp 903–913
117. Mnih V, Kavukcuoglu K, Silver D, Rusu AA, Veness J, Bellemare MG, Graves A, Riedmiller MA, Fidjeland A, Ostrovski G, Petersen S, Beattie C, Sadik A, Antonoglou I, King H, Kumaran D, Wierstra D, Legg S, Hassabis D (2015) Human-level control through deep reinforcement learning. Nature 518(7540):529–533
118. Montavon G, Lapuschkin S, Binder A, Samek W, Müller K-R (2017) Explaining nonlinear classification decisions with deep Taylor decomposition. Pattern Recognit 65:211–222
119. Montavon G, Samek W, Müller K-R (2018) Methods for interpreting and understanding deep neural networks. Digit Signal Process 73:1–15
120. Moraffah R, Karami M, Guo R, Raglin A, Liu H (2020) Causal interpretability for machine learning—problems, methods and evaluation. SIGKDD Explor 22(1):18–33
121. Mothilal RK, Sharma A, Tan C (2020) Explaining machine learning classifiers through diverse counterfactual explanations. In: Hildebrandt M, Castillo C, Celis LE, Ruggieri S, Taylor L, Zanfir-Fortuna G (eds) FAT* '20: conference on fairness, accountability, and transparency, Barcelona, Spain, January 27–30, 2020. ACM, pp 607–617
122. Murdoch WJ, Singh C, Kumbier K, Abbasi-Asl R, Yu B (2019) Interpretable machine learning: definitions, methods, and applications. CoRR, arXiv:1901.04592
123. Nam W-J, Gur S, Choi J, Wolf L, Lee S-W (2020) Relative attributing propagation: interpreting the comparative contributions of individual units in deep neural networks. In: The thirty-fourth AAAI conference on artificial intelligence (AAAI 2020), the thirty-second innovative applications of artificial intelligence conference (IAAI 2020), the tenth AAAI symposium on educational advances in artificial intelligence, (EAAI 2020), New York, NY, USA, February 7–12, 2020. AAAI Press, pp 2501–2508
124. Nguyen AM, Dosovitskiy A, Yosinski J, Brox T, Clune J (2016) Synthesizing the preferred inputs for neurons in neural networks via deep generator networks. In: Lee DD, Sugiyama M, von Luxburg U, Guyon I, Garnett R (eds) Advances in neural information processing systems 29: annual conference on neural information processing systems 2016, December 5–10, 2016, Barcelona, Spain, pp 3387–3395
125. Paszke A, Gross S, Massa F, Lerer A, Bradbury J, Chanan G, Killeen T, Lin Z, Gimelshein N, Antiga L, Desmaison A, Köpf A, Yang EZ, DeVito Z, Raison M, Tejani A, Chilamkurthy S, Steiner B, Fang L, Bai J, Chintala S (2019) Pytorch: an imperative style, high-performance deep learning library. In: Wallach HM, Larochelle H, Beygelzimer A, d'Alché-Buc F, Fox EB, Garnett R (eds) Advances in neural information processing systems 32: annual conference on neural information processing systems 2019, (NeurIPS 2019), December 8–14, 2019, Vancouver, BC, Canada, pp 8024–8035
126. Pearl J et al (2009) Causal inference in statistics: an overview. Stat Surv 6:66
127. Petsiuk V, Das A, Saenko K (2018) RISE: randomized input sampling for explanation of black-box models. In: British machine vision conference 2018 (BMVC 2018), Newcastle, UK, September 3–6, 2018. BMVA Press, p 151 (2018)
128. Pleiss G, Zhang T, Elenberg ER, Weinberger KQ (2020) Identifying mislabeled data using the area under the margin ranking. In: Larochelle H, Ranzato MA, Hadsell R, Balcan M-F, Lin H-T (eds) Advances in neural information processing systems 33: annual conference on neural information processing systems 2020 (NeurIPS 2020), December 6–12, 2020, virtual
129. Plumb G, Al-Shedivat M, Cabrera ÁA, Perer A, Xing EP, Talwalkar A (2020) Regularizing black-box models for improved interpretability. In: Larochelle H, Ranzato MA, Hadsell R, Balcan M-F, Lin H-T (eds) Advances in neural information processing systems 33: annual conference on neural information processing systems 2020 (NeurIPS 2020), December 6–12, 2020, virtual
130. Plumb G, Molitor D, Talwalkar A (2018) Model agnostic supervised local explanations. In: Bengio S, Wallach HM, Larochelle H, Grauman K, Cesa-Bianchi N, Garnett R (eds) Advances in neural information processing systems 31: annual conference on neural information processing systems 2018 (NeurIPS 2018), December 3–8, 2018, Montréal, Canada, pp 2520–2529

131. Plumerault A, Borgne HL, Hudelot C (2020) Controlling generative models with continuous factors of variations. In: 8th International conference on learning representations (ICLR 2020), Addis Ababa, Ethiopia, April 26–30, 2020. OpenReview.net
132. Pope PE, Kolouri S, Rostami M, Martin CE, Hoffmann H (2019) Explainability methods for graph convolutional neural networks. In: IEEE conference on computer vision and pattern recognition (CVPR 2019), Long Beach, CA, USA, June 16–20, 2019. Computer Vision Foundation/IEEE, pp 10772–10781
133. Preuer K, Klambauer G, Rippmann F, Hochreiter S, Unterthiner T (2019) Interpretable deep learning in drug discovery. In: Samek W, Montavon G, Vedaldi A, Hansen LK, Müller K-R (eds) Explainable AI: interpreting, explaining and visualizing deep learning, volume 11700 of lecture notes in computer science. Springer, pp 331–345
134. Puiutta E, Veith EMSP (2020) Explainable reinforcement learning: a survey. In: Holzinger A, Kieseberg P, Tjoa AM, Weippl ER (eds) Machine learning and knowledge extraction—4th IFIP TC 5, TC 12, WG 8.4, WG 8.9, WG 12.9 international cross-domain conference, CD-MAKE 2020, Dublin, Ireland, August 25–28, 2020, proceedings, volume 12279 of lecture notes in computer science. Springer, pp 77–95
135. Puri N, Verma S, Gupta P, Kayastha D, Deshmukh S, Krishnamurthy B, Singh S (2020) Explain your move: understanding agent actions using specific and relevant feature attribution. In: 8th International conference on learning representations (ICLR 2020), Addis Ababa, Ethiopia, April 26–30, 2020. OpenReview.net
136. Rajpurkar P, O'Connell C, Schechter A, Asnani N, Li J, Kiani A, Ball RL, Mendelson M, Maartens G, van Hoving DJ et al (2020) Chexaid: deep learning assistance for physician diagnosis of tuberculosis using chest x-rays in patients with HIV. NPJ Digit Med 6:6
137. Ribeiro MT, Singh S, Guestrin C (2016) "Why should I trust you?": explaining the predictions of any classifier. In: Krishnapuram B, Shah M, Smola AJ, Aggarwal CC, Shen D, Rastogi R (eds) Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining, San Francisco, CA, USA, August 13–17, 2016. ACM, pp 1135–1144
138. Ribeiro MT, Singh S, Guestrin C (2018) Anchors: high-precision model-agnostic explanations. In: McIlraith SA, Weinberger KQ (eds) Proceedings of the thirty-second AAAI conference on artificial intelligence (AAAI-18), the 30th innovative applications of artificial intelligence (IAAI-18), and the 8th AAAI symposium on educational advances in artificial intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2–7, 2018. AAAI Press, pp 1527–1535
139. Ricci F, Rokach L, Shapira B (2011) Introduction to recommender systems handbook. In: Ricci F, Rokach L, Shapira B, Kantor PB (eds) Recommender systems handbook. Springer, pp 1–35
140. Ross AS, Doshi-Velez F (2018) Improving the adversarial robustness and interpretability of deep neural networks by regularizing their input gradients. In: McIlraith SA, Weinberger KQ (eds) Proceedings of the thirty-second AAAI conference on artificial intelligence (AAAI-18), the 30th innovative applications of artificial intelligence (IAAI-18), and the 8th AAAI symposium on educational advances in artificial intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2–7, 2018. AAAI Press, pp 1660–1669
141. Rudin C (2019) Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. Nat Mach Intell 6:66
142. Sabour S, Frosst N, Hinton GE (2017) Dynamic routing between capsules. In: Guyon I, von Luxburg U, Bengio S, Wallach HM, Fergus R, Vishwanathan SVN, Garnett R (eds) Advances in neural information processing systems 30: annual conference on neural information processing systems 2017, December 4–9, 2017, Long Beach, CA, USA, pp 3856–3866
143. Samek W, Binder A, Montavon G, Lapuschkin S, Müller K-R (2017) Evaluating the visualization of what a deep neural network has learned. IEEE Trans Neural Netw Learn Syst 28(11):2660–2673
144. Samek W, Montavon G, Lapuschkin S, Anders CJ, Müller K-R (2021) Explaining deep neural networks and beyond: a review of methods and applications. Proc IEEE 109(3):247–278
145. Selvaraju RR, Cogswell M, Das A, Vedantam R, Parikh D, Batra D (2020) Grad-CAM: visual explanations from deep networks via gradient-based localization. Int J Comput Vis 128(2):336–359
146. Sengupta S, Singh A, Leopold HA, Gulati T, Lakshminarayanan V (2020) Ophthalmic diagnosis using deep learning with fundus images—a critical review. Artif Intell Med 102:101758
147. Seo S, Huang J, Yang H, Liu Y (2017) Interpretable convolutional neural networks with dual local and global attention for review rating prediction. In: Cremonesi P, Ricci F, Berkovsky S, Tuzhilin A (eds) Proceedings of the eleventh ACM conference on recommender systems (RecSys 2017), Como, Italy, August 27–31, 2017. ACM, pp 297–305
148. Serrano S, Smith NA (2019) Is attention interpretable? In: Korhonen A, Traum DR, Màrquez L (eds) Proceedings of the 57th conference of the association for computational linguistics. Association for Computational Linguistics (ACL)

149. Shen Y, Zhou B (2021) Closed-form factorization of latent semantics in gans. In: IEEE conference on computer vision and pattern recognition (CVPR 2021), virtual, June 19–25, 2021. Computer Vision Foundation/IEEE, pp 1532–1540
150. Shrikumar A, Greenside P, Kundaje A (2017) Learning important features through propagating activation differences. In: Precup D, Teh YW (eds) Proceedings of the 34th international conference on machine learning (ICML 2017), Sydney, NSW, Australia, 6–11 August 2017, volume 70 of proceedings of machine learning research (PMLR), pp 3145–3153
151. Silver D, Huang A, Maddison CJ, Guez A, Sifre L, van den Driessche G, Schrittwieser J, Antonoglou I, Panneershelvam V, Lanctot M, Dieleman S, Grewe D, Nham J, Kalchbrenner N, Sutskever I, Lillicrap TP, Leach M, Kavukcuoglu K, Graepel T, Hassabis D (2016) Mastering the game of go with deep neural networks and tree search. Nature 529(7587):484–489
152. Silver D, Schrittwieser J, Simonyan K, Antonoglou I, Huang A, Guez A, Hubert T, Baker L, Lai M, Bolton A, Chen Y, Lillicrap TP, Hui F, Sifre L, van den Driessche G, Graepel T, Hassabis D (2017) Mastering the game of go without human knowledge. Nature 550(7676):354–359
153. Simonyan K, Vedaldi A, Zisserman A (2014) Deep inside convolutional networks: visualising image classification models and saliency maps. In: Bengio Y, LeCun Y (eds) 2nd International conference on learning representations (ICLR 2014), Banff, AB, Canada, April 14–16, 2014, workshop track proceedings
154. Singh A, Sengupta S, Lakshminarayanan V (2020) Explainable deep learning models in medical image analysis. J Imaging 6(6):52
155. Smilkov D, Thorat N, Kim B, Viégas FB, Wattenberg M (2017) Smoothgrad: removing noise by adding noise. CoRR, arXiv:1706.03825
156. Srinivas S, Fleuret F (2019) Full-gradient representation for neural network visualization. In: Wallach HM, Larochelle H, Beygelzimer A, d'Alché-Buc F, Fox EB, Garnett R (eds) Advances in neural information processing systems 32: annual conference on neural information processing systems 2019 (NeurIPS 2019), December 8–14, 2019, Vancouver, BC, Canada, pp 4126–4135
157. Strobelt H, Gehrmann S, Behrisch M, Perer A, Pfister H, Rush AM (2019) Seq2seq-vis: a visual debugging tool for sequence-to-sequence models. IEEE Trans Vis Comput Graph 6:66
158. Strobelt H, Gehrmann S, Pfister H, Rush AM (2018) Lstmvis: a tool for visual analysis of hidden state dynamics in recurrent neural networks. IEEE Trans Vis Comput Graph 6:66
159. Sun Y, Wang S, Li Y-K, Feng S, Chen X, Zhang H, Tian X, Zhu D, Tian H, Wu H (2019) ERNIE: enhanced representation through knowledge integration. CoRR arXiv:1904.09223
160. Sundararajan M, Taly A, Yan Q (2017) Axiomatic attribution for deep networks. In: Precup D, Teh YW (eds) Proceedings of the 34th international conference on machine learning (ICML 2017), Sydney, NSW, Australia, 6–11 August 2017, volume 70 of proceedings of machine learning research (PMLR), pp 3319–3328
161. Swayamdipta S, Schwartz R, Lourie N, Wang Y, Hajishirzi H, Smith NA, Choi Y (2020) Dataset cartography: mapping and diagnosing datasets with training dynamics. In: Webber B, Cohn T, He Y, Liu Y (eds) Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP 2020), Online, November 16–20, 2020. Association for Computational Linguistics, pp 9275–9293
162. Tang J, Wang K (2018) Personalized top-n sequential recommendation via convolutional sequence embedding. In: Chang Y, Zhai C, Liu Y, Maarek Y (eds) Proceedings of the eleventh ACM international conference on web search and data mining (WSDM 2018), Marina Del Rey, CA, USA, February 5–9, 2018. ACM, 565–573
163. Tjoa E, Guan C (2021) A survey on explainable artificial intelligence (XAI): toward medical XAI. IEEE Trans Neural Netw Learn Syst 32(11):4793–4813
164. Toneva M, Sordoni A, des Combes RT, Trischler A, Bengio Y, Gordon GJ (2019) An empirical study of example forgetting during deep neural network learning. In: 7th International conference on learning representations (ICLR 2019), New Orleans, LA, USA, May 6–9, 2019. OpenReview.net
165. Tsipras D, Santurkar S, Engstrom L, Turner A, Madry A (2019) Robustness may be at odds with accuracy. In: 7th International conference on learning representations (ICLR 2019), New Orleans, LA, USA, May 6–9, 2019. OpenReview.net
166. van der Linden I, Haned H, Kanoulas E (2019) Global aggregations of local explanations for black box models. CoRR, arXiv:1907.03039
167. Verma S, Dickerson JP, Hines K (2020) Counterfactual explanations for machine learning: a review. CoRR, arXiv:2010.10596
168. Vinyals O, Babuschkin I, Czarnecki WM, Mathieu M, Dudzik A, Chung J, Choi DH, Powell R, Ewalds T, Georgiev P, Oh J, Horgan D, Kroiss M, Danihelka I, Huang A, Sifre L, Cai T, Agapiou JP, Jaderberg M, Vezhnevets AS, Leblond R, Pohlen T, Dalibard V, Budden D, Sulsky Y, Molloy J, Le Paine T,

Gülçehre Ç, Wang Z, Pfaff T, Wu Y, Ring R, Yogatama D, Wünsch D, McKinney K, Smith O, Schaul T, Lillicrap TP, Kavukcuoglu K, Hassabis D, Apps C, Silver D (2019) Grandmaster level in starcraft II using multi-agent reinforcement learning. Nature 575(7782):350–354
169. Voita E, Talbot D, Moiseev F, Sennrich R, Titov I (2019) Analyzing multi-head self-attention: specialized heads do the heavy lifting, the rest can be pruned. In: Korhonen A, Traum DR, Màrquez L (eds) Proceedings of the 57th conference of the Association for Computational Linguistics (ACL 2019), Florence, Italy, July 28–August 2, 2019, Volume 1: Long Papers. Association for Computational Linguistics, pp 5797–5808
170. Voynov A, Babenko A (2019) RPGAN: gans interpretability via random routing. CoRR, arXiv:1912.10920
171. Voynov A, Babenko A (2020) Unsupervised discovery of interpretable directions in the GAN latent space. In: Proceedings of the 37th international conference on machine learning (ICML 2020), 13–18 July 2020, virtual event, volume 119 of proceedings of machine learning research (PMLR), pp 9786–9796
172. Vu MN, Nguyen TDT, Phan N, Gera R, Thai MT (2019) Evaluating explainers via perturbation. CoRR, arXiv:1906.02032
173. Wachter S, Mittelstadt BD, Russell C (2017) Counterfactual explanations without opening the black box: automated decisions and the GDPR. CoRR, arXiv:1711.00399
174. Wang H, Wang Z, Du M, Yang F, Zhang Z, Ding S, Mardziel P, Hu X (2020) Score-CAM: score-weighted visual explanations for convolutional neural networks. In: 2020 IEEE/CVF conference on computer vision and pattern recognition (CVPR Workshops 2020), Seattle, WA, USA, June 14–19, 2020. Computer Vision Foundation/IEEE, pp 111–119
175. Welinder P, Branson S, Mita T, Wah C, Schroff F, Belongie S, Perona P (2010) Caltech-UCSD birds 200. Technical Report CNS-TR-2010-001, California Institute of Technology
176. Wickramanayake S, Hsu W, Lee M-L (2021) Explanation-based data augmentation for image classification. In: Ranzato MA, Beygelzimer A, Dauphin YN, Liang P, Vaughan JW (eds) Advances in neural information processing systems 34: annual conference on neural information processing systems 2021 (NeurIPS 2021), December 6–14, 2021, virtual, pp 20929–20940
177. Wiegreffe S, Pinter Y (2019) Attention is not not explanation. In: Inui K, Jiang J, Ng V, Wan X (eds) Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP). Association for Computational Linguistics
178. Woo S, Park J, Lee J-Y, Kweon IS (2018) CBAM: convolutional block attention module. In: Ferrari V, Hebert M, Sminchisescu C, Weiss Y (eds) Computer vision—ECCV 2018—15th European conference, Munich, Germany, September 8–14, 2018, proceedings, Part VII, volume 11211 of lecture notes in computer science. Springer, pp 3–19
179. Xu G, Duong TD, Li Q, Liu S, Wang X (2020) Causality learning: a new perspective for interpretable machine learning. CoRR, arXiv:2006.16789
180. Yang C, Shen Y, Zhou B (2021) Semantic hierarchy emerges in deep generative representations for scene synthesis. Int J Comput Vis 129(5):1451–1466
181. Yang M, Kim B (2019) Benchmarking attribution methods with relative feature importance
182. Yao Y, Chen T, Xie G-S, Zhang C, Shen F, Wu Q, Tang Z, Zhang J (2021) Non-salient region object mining for weakly supervised semantic segmentation. In: IEEE conference on computer vision and pattern recognition (CVPR 2021), virtual, June 19–25, 2021. Computer Vision Foundation/IEEE, pp 2623–2632
183. Yeh C-K, Hsieh C-Y, Suggala AS, Inouye DI, Ravikumar P (2019) On the (in)fidelity and sensitivity of explanations. In: Wallach HM, Larochelle H, Beygelzimer A, d'Alché-Buc F, Fox EB, Garnett R (eds) Advances in neural information processing systems 32: annual conference on neural information processing systems 2019 (NeurIPS 2019), December 8–14, 2019, Vancouver, BC, Canada, pp 10965–10976
184. Ying Z, Bourgeois D, You J, Zitnik M, Leskovec J (2019) Gnnexplainer: generating explanations for graph neural networks. In: Wallach HM, Larochelle H, Beygelzimer A, d'Alché-Buc F, Fox EB, Garnett R (eds) Advances in neural information processing systems 32: annual conference on neural information processing systems 2019 (NeurIPS 2019), December 8–14, 2019, Vancouver, BC, Canada, pp 9240–9251
185. Yuan T, Li X, Xiong H, Cao H, Dou D (2021) Explaining information flow inside vision transformers using Markov chain. In: Neural information processing systems XAI4Debugging workshop
186. Zagoruyko S, Komodakis N (2017) Paying more attention to attention: improving the performance of convolutional neural networks via attention transfer. In: 5th International conference on learning representations (ICLR 2017), Toulon, France, April 24–26, 2017, conference track proceedings. OpenReview.net

187. Zhang C, Bengio S, Hardt M, Recht B, Vinyals O (2021) Understanding deep learning (still) requires rethinking generalization. Commun ACM 64(3):107–115
188. Zhang H, Cissé M, Dauphin YN, Lopez-Paz D (2018) Mixup: beyond empirical risk minimization. In: 6th International conference on learning representations (ICLR 2018), Vancouver, BC, Canada, April 30–May 3, 2018, conference track proceedings. OpenReview.net
189. Zhang J, Bargal SA, Lin Z, Brandt J, Shen X, Sclaroff S (2018) Top-down neural attention by excitation backprop. Int J Comput Vis 126(10):1084–1102
190. Zhang Q, Cao R, Shi F, Wu YN, Zhu S-C (2018) Interpreting CNN knowledge via an explanatory graph. In: McIlraith SA, Weinberger KQ (eds) Proceedings of the thirty-second AAAI conference on artificial intelligence (AAAI-18), the 30th innovative applications of artificial intelligence (IAAI-18), and the 8th AAAI symposium on educational advances in artificial intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2–7, 2018. AAAI Press, pp 4454–4463
191. Zhang Q, Wu YN, Zhu S-C (2018) Interpretable convolutional neural networks. In: 2018 IEEE conference on computer vision and pattern recognition (CVPR 2018), Salt Lake City, UT, USA, June 18–22, 2018. Computer Vision Foundation/IEEE Computer Society, pp 8827–8836
192. Zhang Q, Yang Y, Ma H, Wu YN (2019) Interpreting cnns via decision trees. In: IEEE conference on computer vision and pattern recognition (CVPR 2019), Long Beach, CA, USA, June 16–20, 2019. Computer Vision Foundation/IEEE, pp 6261–6270
193. Zhang S, Yao L, Sun A, Tay Y (2019) Deep learning based recommender system: a survey and new perspectives. ACM Comput Surv 52(1):51–538
194. Zhang T, Zhu Z (2019) Interpreting adversarially trained convolutional neural networks. In: Chaudhuri K, Salakhutdinov R (eds) Proceedings of the 36th international conference on machine learning (ICML 2019), 9–15 June 2019, Long Beach, California, USA, volume 97 of proceedings of machine learning research (PMLR), pp 7502–7511
195. Zhang Y, Chen X (2020) Explainable recommendation: a survey and new perspectives. Found Trends Inf Retr 14(1):1–101
196. Zhao G, Zhou B, Wang K, Jiang R, Xu M (2018) Respond-CAM: analyzing deep models for 3d imaging data by visualizations. In: Frangi AF, Schnabel JA, Davatzikos C, Alberola-López C, Fichtinger G (eds) Medical image computing and computer assisted intervention—MICCAI 2018—21st international conference, Granada, Spain, September 16–20, 2018, proceedings, Part I, volume 11070 of lecture notes in computer science. Springer, pp 485–492
197. Zhou B, Khosla A, Lapedriza À, Oliva A, Torralba A (2016) Learning deep features for discriminative localization. In: 2016 IEEE conference on computer vision and pattern recognition, (CVPR 2016), Las Vegas, NV, USA, June 27–30, 2016. IEEE Computer Society, pp 2921–2929

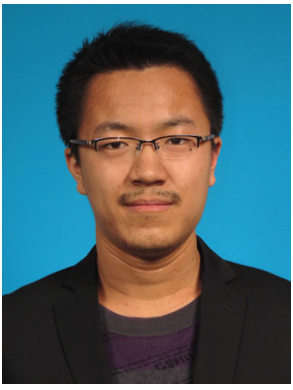


Xuhong Li is a staff researcher at Big Data Laboratory of Baidu Research. He received his bachelor degree and master degree from Beihang University, China and his Ph.D. degree from University of Technology of Compiègne, France in 2019. He is interested in and working on Explainable AI and Transfer Learning, as well as Self-Supervised Learning and Multi-Modal Learning, for both computer vision and natural language processing applications. He has served regularly among the program committees for top international machine learning conferences, such as ICML and Neurips.

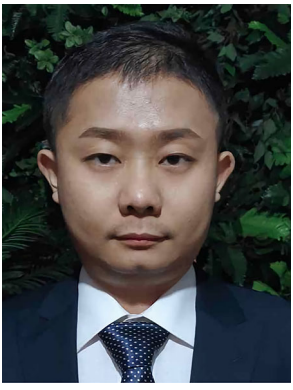
Haoyi Xiong received the Ph.D. degree in Electrical and Computer Engineering from Télécom SudParis jointly with Université Pierre et Marie Curie, France, in 2015. From 2016 to 2018, he was a Tenure-Track Assistant Professor with the Department of Computer Science, Missouri University of Science and Technology, Rolla, MO and a Postdoc at University of Virginia, Charlottesville VA (2015–2016). He joined Baidu Research, Beijing, China in 2018, where he is currently a Principal R&D Architect. He also serves as a Graduate Faculty Scholar affiliated to University of Central Florida, Orlando FL. His current research interests include machine learning and ubiquitous computing. He has published more than 70 papers in top computer science conferences and journals, including ICML, UbiComp, KDD, RTSS, ICLR, ICCV, AAAI, IJCAI, IEEE/ACM Transactions and received more than 3400 Google Scholar Citations. He is a recipient of IEEE UIC'12 Best Paper Award (2012), CNRS Samovar Outstanding Ph.D Thesis Runner-up (2015), the 1st Prize of Science & Technology Advancement Award from Chinese Institute of Electronics (2019), IEEE TCSC Early Career Researcher Award (2020). He is a Senior Member of IEEE, a Member of ACM and Sigma Xi.

Xingjian Li received the B.S. degree in microelectronics from Tsinghua University, in 2008, the M.S. degree in computer science and technology from the Institute of Computing Technology, Chinese Academy of Sciences, in 2011. Currently, he is a senior researcher at Big Data Lab, Baidu Research. His research interests include deep learning, transfer learning and semi-supervised learning.

Xuanyu Wu is a Machine Learning Engineer at Alibaba Group, working on the business applications of machine learning and deep learning. The primary focus is on recommender systems. He obtained his master degree at University of Pennsylvania in 2022 and bachelor degrees at University of California, San Diego in 2020.

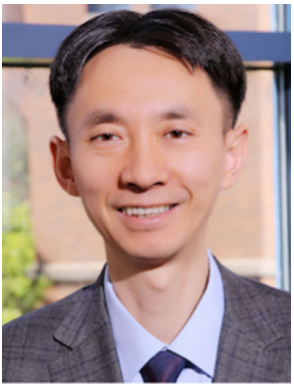


Xiao Zhang received the bachelor's and master's degrees from the Department of Electronic Engineering, Tsinghua University in 2015 and 2018. He is currently a Ph.D. candidate at the Multimedia Signal and Intelligent Information Processing Lab (MSIIP) of the Department of Electronic Engineering, Tsinghua University. His research interests include continual learning and representation learning for natural language processing.



Ji Liu is a staff researcher with the Big Data Laboratory, Baidu Research, Beijing, China. He was a software engineer at Murex and Post-Doctoral research fellow with Inria and LIRMM, University of Montpellier, France. Previously, he was a Ph.D. candidate in the Microsoft Research Inria Joint Centre with the Inria Zenith team. His research interests include federated learning, distributed machine learning, scientific workflows, big data and distributed system. He has published several papers in top international journals and conferences, such as NeurIPS, AAAI, IJCAI, KDD, TKDD, TITS, and TKDE, and is a co-author of the book "Data-Intensive Workflow Management For Clouds and Data-Intensive and Scalable Computing Environments" published by Morgan Claypool in 2019. He obtained a Ph.D. degree from University of Montpellier in 2016, a Master degree from Telecom SudParis in 2013, and a BsC degree from Xidian University in 2011.

Jiang Bian received the Ph.D. degree of Computer Engineering in University of Central Florida, Orlando, FL. He received the B.Eng. degree of Logistics Systems Engineering in Huazhong University of Science and Technology, Wuhan, China, in 2014, and the M.Sc. degree of Industrial Systems Engineering in University of Florida at Gainesville, FL, in 2016. He is currently with the Big Data Laboratory, Baidu Research, Beijing, China. His research interests include Ubiquitous Computing, AutoDL, IoT, and Intelligent Systems.



Dejing Dou is the Head of Big Data Lab (BDL) and Business Intelligence Lab (BIL) at Baidu Research. He is also a full Professor (on leave) from the Computer and Information Science Department at the University of Oregon. He received his bachelor degree from Tsinghua University, China in 1996 and his Ph.D. degree from Yale University in 2004. His research areas include artificial intelligence, data mining, data integration, NLP, and health informatics. Dejing Dou has published more than 150 research papers, some of which appear in prestigious conferences and journals like AAAI, IJCAI, ICML, NeurIPS, ICLR, KDD, ICDM, ACL, EMNLP, CVPR, ICCV, CIKM, ISWC, TKDD, JIIS, and JoDS, with more than 5000 Google Scholar citations. His DEXA'15 paper received the best paper award. His KDD'07 paper was nominated for the best research paper award. His COLING'18 paper was Area Chair Favorites (excellent). He is on the Editorial Boards of Journal on Data Semantics, Journal of Intelligent Information Systems, and PLOS ONE. He is a Editor-in-Chief of AIMS Electronic Research Archive. He has been serving as program committee members for major international conferences and as program co-chairs for five of them. He has received over 5 million PI research grants from the NSF and the NIH. Dejing Dou is a senior member of ACM and IEEE.

Authors and Affiliations

Xuhong Li[1] · Haoyi Xiong[1] · Xingjian Li[1] · Xuanyu Wu[2] · Xiao Zhang[3] · Ji Liu[1] · Jiang Bian[1] · Dejing Dou[1,4]

Xuhong Li
lixuhong@baidu.com

Haoyi Xiong
xionghaoyi@baidu.com

Xingjian Li
lixingjian@baidu.com

Xuanyu Wu
xuanyuwu@seas.upenn.edu

Xiao Zhang
xzhang19@mails.tsinghua.edu.cn

Ji Liu
liuji04@baidu.com

Jiang Bian
bianjiang03@baidu.com

1 Baidu Research, Baidu Inc., Beijing, China

2 School of Engineering and Applied Science, University of Pennsylvania, Philadelphia, PA, USA

3 Department of Electronics and Information Engineering, Tsinghua University, Beijing, China

4 Computer and Information Science Department, University of Oregon, Eugene, OR, USA