Causal Interpretability for Machine Learning
- Problems, Methods and Evaluation

Raha Moraffah∗, Mansooreh Karami∗, Ruocheng Guo∗, Adrienne Raglin†, Huan Liu∗

∗Computer Science & Engineering, Arizona State University, Tempe, AZ, USA

†Army Research Lab, USA

∗{rmoraffa, mkarami, rguo12, huanliu}@asu.edu, †adrienne.raglin2.civ@mail.mil

ABSTRACT

Machine learning models have had discernible achievements in a myriad of applications. However, most of these models are black-boxes, and it is obscure how the decisions are made by them. This makes the models unreliable and untrustworthy. To provide insights into the decision making processes of these models, a variety of traditional interpretable models have been proposed. Moreover, to generate more human-friendly explanations, recent work on interpretability tries to answer questions related to causality such as "Why does this model makes such decisions?" or "Was it a specific feature that caused the decision made by the model?". In this work, models that aim to answer causal questions are referred to as causal interpretable models. The existing surveys have covered concepts and methodologies of traditional interpretability. In this work, we present a comprehensive survey on causal interpretable models from the aspects of the problems and methods. In addition, this survey provides in-depth insights into the existing evaluation metrics for measuring interpretability, which can help practitioners understand for what scenarios each evaluation metric is suitable.

Keywords

Interpratablity, explainability, causal inference, counterfactuals, machine learning

1. INTRODUCTION

With the surge of machine learning in critical areas such as healthcare, law-making and autonomous cars, decisions that had been previously made by humans are now made automatically using these algorithms. In order to ensure the reliability of such decisions, humans need to understand how these decisions are made. However, machine learning models are usually inherently black-boxes and do not provide explanations for *how* and *why* they make such decisions. This has become especially problematic when recent work shows that the decisions made by machine learning models are sometimes biased and enforce inequality [69]. For instance, Angwin et al. [4] demonstrates that predictions made by Correctional Offender Management Profiling for Alternative Sanctions (COMPAS), which is a widely used criminal risk assessment tool, shows racial biases. With recent regulations such as European Union's "Right to Ex-

planation" [32] and AI call for diversity and inclusion [9], interpretable models which are capable of explaining the decisions they made are necessary. Moreover, recent research shows that machine learning models, especially deep neural networks, can be easily fooled into predicting a specific class label for an image when its pixel values are under minimal perturbations [30, 74, 80]. Such results imply that machine learning models suffer from the risk of making unexpected decisions. Understanding decisions of machine learning models and the process leading to decision making can help us understand the rules the models use to make their decisions and therefore, prevent potential unexpected situations from happening. More specifically, through interpretable machine learning models, we aim to guarantee that (a) decisions made by machine learning models comply with the rules toward social good; (b) the classifier does not pick up the biases in the data and the decisions made are compatible with human understandings.

Previously, various frameworks have been proposed to generate explanations for machine learning algorithms. These algorithms can be mainly divided into two categories, (1) algorithms that are inherently interpretable, which includes the models that generate explanations at training time [107]; (2) post-hoc interpretations that refer to the model that generate explanations for already made decisions [75, 85, 47]. Henceforth, these models are referred to as traditional interpretable models.

In this work, we focus on causal interpretable models that can explain their decisions through what decisions would have been made if they had been under alternative situations (e.g., being trained with different inputs, model components or hyperparameters). Note that traditional interpretable models are unable to answer such questions about decision making under alternative situations, although they can explain how and why a decision is made by an existing model on an observed instance. For instance, in the case of credit applications, to impose fairness on the decision making process, we may need to answer questions such as Did the protected features (e.g., race and gender etc.) cause the system to reject the application of the i-th applicant?" and "If the i-th applicant had different protected features, would the system still make the same decision?" In other words, in order to make the explanations more understandable and useful for humans, we need to ask questions such as "Why did the classifier make this decision instead of another?", "What would have happened to this decision of a classifier had we had a different input to it?", or "Was it feature X that caused decision Y?". Traditional interpretabil-

ity frameworks which only consider correlations are not capable of generating such explanations. This is due to the fact that these frameworks cannot estimate how altering a feature or a component of a model would change the predictions made by the rest of the model or the predicted labels on the data samples. Therefore, in order to answer such questions about both data samples and models, counterfactual analysis needs to be leveraged. Counterfactual analysis is a concept from the causal inference literature [25]. In counterfactual analysis, we aim to infer the output of a model in imaginary scenarios that we have not observed or cannot observe. Recently, counterfactual analysis and causal inference have gained a lot of attention from the interpretable machine learning field. Research in this area has mainly focused on generating counterfactual explanations from both the data perspective [34, 76] as well as the components of a model [77, 38].

Existing surveys on interpretable machine learning focus on the traditional methods and do not discuss the existing methods from a causal perspective. In this survey, we present commonly used definitions for interpretability, discuss interpretable models from a causal perspective and provide guidelines for evaluating these methods. More specifically, in Section 2, we first provide different definitions for interpretability. We then briefly introduce the existing methods on traditional interpretablity and present different types of interpretable models in this category (Section 2.2). Section 3 discusses concepts from causal inference, which are used in this survey. In section 4, we provide an overview of existing works on causal interpretability. We also compare the proposed models for both traditional and causal models from different perspectives to provide insights on advantages and disadvantages of each type of interpretability. Section 5 provides detailed guidelines on the experimental settings such as commonly used datasets and evaluation metrics for both traditional and causal approaches. We then discuss evaluation metrics specifically used for causal methods in more detail and provide different scenarios for which these metrics can be used. Since the evaluation of causal interpretable models is a challenging task, these guidelines can be helpful for future research in this area and can be used to evaluate approaches with similar characteristics. In addition, they can also be used to create new evaluation metrics for the approaches with different functionalities.
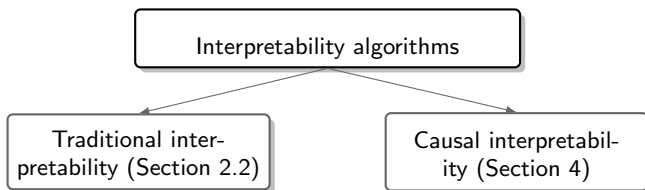

Figure 1: Main categories for Interpretable frameworks

2. AN OVERVIEW OF INTERPRETABILITY

In this section, we present an overview of existing definitions for interpretability. Miller et al. [70] suggest that interpretability is the degree to which a human can understand the cause of a decision. Kim et al. [48] propose that interpretability is the degree to which a human can consistently predict the model's decisions. Doshi-Velez et al. [17] define interpretability as the ability to explain in intelligible ways to a human. Gilpin et al. [27] take a step further and define interpretability as a part of explainability. They state that explainable models are those that summarize the reasons for neural network behaviors, gain the trust of the users, or generate insights into the causes of their decisions while interpretable models may not be able to describe the operation of a system in an accurate way [1]. Pearl [84] claims that tasks such as explainability require a causal model of the environment and cannot be handled at the level of association.

2.1 Interpetability in Machine Learning

Interpretable machine learning has been widely explored and discussed in previous literature. However, to the best of our knowledge, there is no comprehensive review on causal interpretability models. For instance, Lipton [59] discusses the motivation behind creating interpretable models and categorizes interpretable models into two main categories: transparent models and post-hocs. Doshi-velez et al. [17] provide a definition of model interpretability and evaluation criteria. However, this review only proposes definitions and evaluations that are used for traditional interpretability of models and does not cover causal and counterfactual questions. Gilpin et al. [27], explain fundamental concepts of explainability and use them to classify the literature on interpretable models. Zhang and Zhu [112] review the existing interpretable models proposed for deep models used in visual domains. Du et al. [18] provide a comprehensive survey of existing interpretable methods and discuss issues that should be considered in future work. It is worth mentioning that none of the existing work discussed interpretable models from a causal perspective. In this work, we first introduce the state-of-the-art research in traditional interpretability (Sec. 2.2) and then give a detailed survey on causal interpretable models (Sec. 4). Figure 1 shows an overview of interpretable models and their classification.

2.2 Traditional Interpretablity

Before proceeding with the detailed review of the methodologies in causal interpretable models, we provide an overview of existing state-of-the-art methods in traditional machine learning. We categorize traditional models into two main categories:

- Inherently interpretable models: Models that generate explanations in the process of decision making or while being trained.

- Post-hoc interpretability: Generating explanations for an already existing model using an auxiliary model. Example-based interpretablity also falls into this category. In example-based interpretablity, we are looking for examples from the dataset which explain the model's behavior the best.

2.2.1 Interpretable Models

A machine learning model can be designed to include explanations embedded as part of their architecture or output interpretable decisions as part of their training process.

---

[1] In this survey we use the words interpretable and explainable interchangeably.

Most of these models are created in application of the deep neural network. In this section, we present common interpretable models in the literature.

Decision Trees. These methods make use of a tree-structured framework in which each internal node checks whether a condition on a feature is satisfied or not while the leaf nodes show the final predictions (class labels). A decision infers the label of an instance by starting from the root and tracing a path till a leaf node is reached, which can be interpreted as an *if..then..* rule. An example is illustrated in Figure 2.
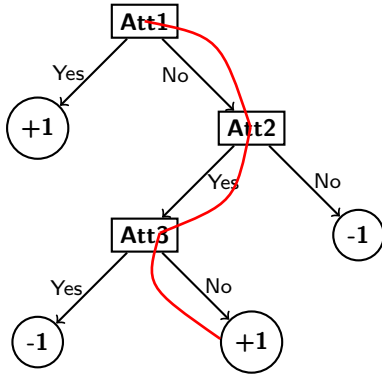

Figure 2: An example of a decision tree with positive and negative class (binary) and three attributes. The red path has a decision rule, if $\neg Att1 \land Att2 \land \neg Att3 \Rightarrow +1$

Rule-Based Models. Rule-based classifiers also create explanations that are interpretable for humans. These classifiers use a collection of *if..then..* rules to infer the class labels. In a sense, rule-based classifiers can be considered as the text representation of the decision trees (i.e., a set of decision rules). However, there are some key differences. Rule-based models can have rules that are not mutually exclusive (i.e., two or more rules might trigger by the same record), not exhausted (i.e., a record may not trigger any rules) and ordered (i.e., the rule set is ordered based on their priority) [95].

Linear Regression. Another common method known to be interpretable is Linear Regression. Linear Regression models the linear relation between a dependent variable and a set of explanatory variables (features). The weight of each feature represents the mean change in the prediction given a one unit increase of the feature. Accordingly, it is reasonable to think that the features with larger weights has more effect on the final result. However, different types of variables (e.g., categorical data vs numerical features) have different scales. This makes it difficult to interpret the effect of each feature. Fortunately, there are several methods that can be used to find the importance of a feature in a linear regression such as t-statistics and chi-square score [57].

The aforementioned methods are restricted by users' limitations (i.e., human understanding). With the increase in the number of features, these models become more and more complex; for example, decision trees become much deeper, and the number of the rules increase in the rule sets. This makes comprehending the prediction of these models difficult for humans [89]. Below, we discuss recent inherently interpretable models which are designed for more sophisti-

cated scenarios.

Attention Networks. Attention networks have been successful in various highly-impactful tasks such as graph embedding [99] and machine translation [98, 6]. These models are widely known not only for their improved performance over previous methods but also for their capability to show which input features or learned representations are more important for making a specific prediction. Yang et al. [107] use a hierarchical attention network in document classification to capture the informative words as well as the sentences that have a significant role in the decision. This is because the same word or sentence may be differentially important in different contexts. Attention networks also proved to be a useful tool in visual question answering applications, which require a joint image-text understanding to answer a question about the image [104, 64, 103, 65]. Yang et al. [106] propose a Stacked Attention Network (SAN) that uses two attention layers to infer the answer progressively. While the first attention layer focuses on all referred concepts in the question, the higher-level layer provides a sharper attention distribution to highlight regions that are more relevant to the answer.

Disentangled Representation Learning. One goal of representation learning is to break down the features into the independent latent variables that are highly correlated with meaningful patterns [29]. In traditional machine learning, approaches such as PCA [44], ICA [42] and spectrum analysis [100] are proposed to discover disentangled components of data. Recently, deep latent-variable models such as VAE [50], InfoGAN [13] and β-VAE [40] were developed to learn disentangled latent variables through variational inference. For example, in empirical studies, it is shown that β-VAE and InfoGAN can learn interpretable factorized latent variables of human face images such as azimuth, hairstyle and emotion [40].

2.2.2 Post-hoc Interpretability

Post-hoc interpretable methods aim to explain the decision-making process of the black-box models after they are trained. These methods map an abstract concept used in a trained machine learning model into a domain that is understandable by humans such as a block of pixels or a sequence of words. Following are the widely known post-hoc methods.

Local Explanations. Local Interpretable Model-Agnostic Explanations (LIME) [89] is a representative and pioneer framework that generates local explanations of black-box models. LIME approximates the prediction of any black-box via local surrogate interpretable models. LIME selects an instance to explain by perturbing it around its neighborhood (i.e., eliminating patches of pixels or zeroing out the values of some features). These samples are then fed to the complex model for labeling and then it will be weighted based on their proximity to the original data. Finally, LIME learns an interpretable model on the weighted perturbed data and their associated labels to create the explanations. It is worth noting that LIME is a fast approximation of a broader approach named SHAP [66] that measures feature importance.

Saliency Maps. Originally introduced by Simonyan et al. [93] as "image-specific class saliency maps", saliency maps highlight pixels of a given input image that are mostly involved in deciding a particular class label for the image. To extract those pixels, the derivative of the weight vector is found by a single backpropagation pass (deconvolution).

The magnitude of the derivative shows the importance of each pixel for the class score. Similar concepts were used by other researchers to deconvolve the prediction and show the locations of the input image that strongly impacts the activation of the neurons [109, 94, 91]. While these methods belong to a popular class of tools for interpretability, Adebayo et al. [2] and Ghorbani et al. [26] suggest that relying on visual assessment is not adequate and can be misleading.

Example-Based Explanations. As proved in education [87] and psychology domains [1], learning from experiences and examples are promising tools to explain complex concepts. In these methods, a certain example is selected from the dataset to represent the model's prediction (e.g., k-nearest neighbor) or the distribution of the data. It is worth mentioning that example-based explanations should not be confused with those explanations that perturb *features* in the dataset [85]. Although using prototypes as the representation of data has shown to be effective in the human learning process [1], Kim et al. [47] use a method called Maximum Mean Discrepancy (MMD) to capture a more complex distribution of the data. This method uses some instances as *criticisms* to explain which prototypes are *not* captured by the model to improve the interpretability of the black-boxes. Gurumoorthy et al. [37] extend this method and designed a fast prototype selection algorithm called *ProtoDash* to not only select the prototypes and criticism instances, but also output non-negative weights indicating their importance.

Influence Functions. To track the impact of a training sample on the prediction of a machine learning model, one can simply modify an example or delete it (leave-one-out), retrain the model, and observe the effect. However, this approach can be extremely expensive. To alleviate the issue, influence functions, a classic method from the robust statistics literature, can be used. Koh and Liang [52] proposed a second-order optimization technique to approximate these influence functions. They verified their technique with different assumptions on the empirical risk ranging from being strictly convex and twice-differentiable to non-convex and non-differentiable losses.

Suppose $\hat{y}(x_t, \hat{\theta})$ is the model's prediction for the sample x_t with an optimal parameter $\hat{\theta}$. Lets $\hat{y}(x_t, \hat{\theta}_{-z})$ be the prediction on the sample x_t when the training sample z was removed while the model's optimal parameter is $\hat{\theta}_{-z}$. The influence function tries to approximate the difference between the two predictions, $\hat{y}(x_t, \hat{\theta}) - \hat{y}(x_t, \hat{\theta}_{-z})$, without retraining the model.

The same authors [51] also investigate the effect of removing large groups of training points in large datasets on the accuracy of influence functions. They find out that the approximation computed by the influence functions are correlated with the actual effect. Inspired by this work, Cheng et al. [14] propose an explanation method, *Fast Influence Analysis*, that employs influence functions on Latent Factor Models to resolve the lack of interpretability of the collaborative filtering approaches for recommender systems.

Feature Visualization. Another way of describing what the model has learned is feature visualization. Most methods in this category deal with image inputs. Erhan et al. [20] present an optimization technique called *activation maximization* to visualize what a neuron computes in an arbitrary layer of deep neural network. Let $\hat{\theta}$ be the learned fixed parameters after training and $h_{ij}(\hat{\theta}, x)$ be the activation of neuron i in layer j, the learned image for that neuron can be calculated by solving the following optimization problem,

$$x^* = \arg\max_x h_{ij}(\hat{\theta}, x), \text{ subject to } ||x||_2 = 1 \qquad (1)$$

This equation will find out ultimate input patterns (x^*) which triggers a neuron to get activated. Despite this method being used as a tool in providing explanations for higher-layer features [55, 79, 75], it has been reported that due to the complexity of the input distribution, some returned images might contain optical illusions [78, 20].

Explaining by Base Interpretable Models. In Section 2.2.1 we discussed base models such as decision tree, rule-based and linear regression, that are known to be interepretable. Following, we will introduce some works that utilize these algorithms to explain a more sophisticated framework. Craven and Shavlik [16] are one of the first to use tree-structured representations to approximate neural networks. Since their model is independent of the network architecture and training algorithm, it can be generalized to a wide variety of models. Their method, *TREPAN*, is similar to CART and C4.5 and uses a gain ratio criterion to evaluate the potential splits, but expands the tree based on a node that increases the fidelity of the extracted tree to the network. Inspired by *TREPAN*, Boz [10] propose a method called *DECTEXT* to extract a decision tree that mimics the behavior of a trained Neural Network. In their method, they propose a new splitting technique, a new discretization method, and a novel pruning procedure. With these modifications, the proposed method can handle continuous features, optimize fidelity and minimize the tree size. A technique called distillation [41] can also be used to fully understand why a specific answer is returned for a particular example. Frosst and Hinton [24] answer this question by creating a model in the form of soft decision tree and examine all the learned filters from the root of the tree to the classification's leaf node. Zhang et al. [111] adopt the same concept but explained the network knowledge at a human-interpretable semantic level and also showed how much each filter contributes to the prediction.

The MofN algorithm [96] is one of the well-known methods that is used to extracts symbolic rules from trained neural networks. This method clusters the links based on the weights and eliminates those groups that unlikely to have any impact on the consequent. It then forms rules that are the sum of the weighted antecedents with regard to the bias. Authors also report experiments on the fidelity of the model and the comprehensibility of the set rules and the individual rules.

Lou et al. [62] use a generalized version of linear regression called *generalized additive models* (GAM) in the form of $g(y) = \sum f_i(x_i) = f_1(x_1) + ... + f_n(x_n)$ to interpret the contribution of each predictor for different classifiers or regression models. $g(.)$ is a link function that controls whether we want to describe the model as an additive model (regression by setting $g(y) = y$) or generalized additive model (classification by setting it to a logistic function). $f(.)$ is a shape function that quantifies the impact of each individual feature. This gives the ability to interpret spline models and tree-based shape functions such as single trees, bagged trees, boosted trees and boosted-bagged trees. Due to the model not considering the interactions between the features, there is a significant gap in terms of accuracy between these models and complex models. To fill this gap, the same authors propose a method named *Gen-*

eralized Additive Models plus Interactions (GA^{2}Ms) in the form of $g(y) = \sum f_i(x_i) + \sum f_{ij}(x_i, x_j)$ which takes into account the two-dimensional interactions that still can be interpretable as heat maps [63]. Two case studies are conducted on real healthcare problems on predicting pneumonia risks by using GA^{2}Ms. These studies uncover new patterns that are ignored by state-of-the-art complex models while still hitting their accuracy [11].

3. CAUSAL INFERENCE

In this section, we briefly review the concepts from causal inference used in this paper for causal interpretable models. In their paper, Guo et al. [36] provide a comprehensive review of existing causal inference methods and definitions.

DEFINITION 1 (STRUCTURAL CAUSAL MODELS). *A 4-tuple variable $M(X, U, f, P_u)$ where X is a finite set of endogenous variables, usually the observable variables, U denotes a finite set of exogenous variables which usually account for unobserved or noise variables, f is a set of function $\{f_1, f_2, ..., f_n\}$ where each function represents a causal mechanism such that $\forall x_i \in X, x_i = f_i(Pa(x_i), u_i)$ and $Pa(x_i)$ is a subset of $(X \setminus \{x_i\}) \cup U$ and P_u is a probability distribution over U is called a Structural Causal Model (SCM) or Structural Equation Model (SEM)[82].*

DEFINITION 2 (CAUSAL BAYESIAN NETWORK). *To represent an SCM $M(X, U, f, P_u)$, a directed graphical model $G(V, E)$ is used. V is the set of endogenous variables X and E denotes the causal mechanisms. This indicates for each causal mechanism $x_i = f_i(Pa(x_i), u_i)$, there exists a directed edge from each node in the parent set $Pa(x_i)$ to x_i. The entire graph representing this SCM is called a Causal Bayesian Network (CBN).*

DEFINITION 3 (AVERAGE CAUSAL EFFECT). *The Average Causal Effect (ACE) of a binary random variable x (treatment) on another random variable y (outcome) is defined as:*

$$ACE = \mathbb{E}[y|do(x=1)] - \mathbb{E}[y|do(x=0)], \qquad (2)$$

Where $do(.)$ operator denotes the corresponding interventional distribution defined by the SCM or CBN.

4. CAUSAL INTERPRETABLITY

In this section, we discuss the state-of-the-art frameworks on causal interpretability. These frameworks are particularly needed since objective functions of machine learning models only capture correlations and not real causes. Therefore, these models might cause problems in real-world decision making, such as making policies related to smoking and cancer. Moreover, training data used to train these models might not perfectly represent the environment; and the train and the test sets might also have different distributions. A causal interpretable model can help us understand the real causes of decisions made by machine learning algorithms, improve their performance, and prevent them from failing in unexpected circumstances.

Pearl [83] introduces different levels of said interpretability and argues that generating counterfactual explanations is the way to achieve the highest level of interpretability. Below are those levels of interpretability and their definitions:

- Statistical (associational) interpretability: Aims to uncover statistical associations by asking questions such as "How would seeing x change my belief in y?"

- Causal interventional interpretability: Is designed to answer "What if" questions.

- Counterfactual interpretability: Is the highest level of interpretability, which aims to answer "Why" questions.

Traditional interpretability mainly focuses on the statistical interpretability, whereas causal interpretability aims to answer questions associated with the causal interventional interpretability and counterfactual interpretability. In the following, we provide an extensive review of existing work on causal interpretability. We classify the existing works in this field into four main categories:

1. Causal inference and model-based interpretation: In this category, methods explain the causal effect of a model component on the final decision.

2. Causal inference and example-based interpretation: All Methods in this category aim to generate counterfactual explanations for alternate situations and scenarios.

3. Causal inference and fairness: Lipton [59] explains that interpretable models are often indispensable to guarantee fairness. Motivated by this, we provide an overview of the state-of-the-art methods on causal fairness.

4. Causal inference as guarantee for interpretability: In this category, we review methods which leverage interpretability as a tool to verify causal assumptions and relationships. We also discuss the scenarios, where causal inference can be used to guarantee the interpretability of a machine learning model.

In the following, we discuss each category in detail.

4.1 Causal Inference and Model-based Interpretation

Recently, causality has gained increasing attention in explaining machine learning models [12, 38]. These approaches are usually designed to explain the role and importance of each component of a machine learning model on its decisions with concepts from the causality. For instance, one way to explain the role of a neuron on the decision of a neural network is to estimate the ACE of the neuron on the output [12, 81]. Traditional interpretable models cannot answer vital questions for understanding machine learning models. For instance, traditional machine interpretability frameworks are not capable to answer causal questions such as "What is the impact of the n-th filter of the m-th layer of a deep neural network on the predictions of the model?" which are helpful and required for understanding a neural network model. Furthermore, despite being simple and intuitive, performing ablation testing (i.e., removing a component of the model and retraining it to measure the performance for a fixed dataset) is computationally expensive and impractical. To address these problems, causal interpretability frameworks have been proposed. These frameworks are mainly designed to explain the importance of each component of a

deep neural network on its predictions by answering counterfactual questions such as "What would have happened to the output of the model had we had a different component in the model?". These types of questions are answered by borrowing some concepts from the causal inference literature. The main idea is to model the structure of the DNN as an SCM and estimate the causal effect of each component of the model on the output by performing causal reasoning. Narendra et al. [77] consider the DNN as an SCM, apply a function on each filter of the model to obtain the targeted value such as variance or expected value of each filter and reason on the obtained SCM. Harradon et al. [38] further suggest that in order to have an effective interpretability, having a human-understandable causal model of DNN, which allows different kinds of causal interventions, is necessary. Based on this hypothesis, the authors propose an interpretability framework, which extracts human-understandable concepts such as eyes and ears of a cat from deep neural networks, learns the causal structure between the input, output and these concepts in an SCM and performs causal reasoning on it to gain more insights into the model. Chattopadhyay et al. [12] propose an attribution method based on the first principle of causality, particularly SCMs and $do(\cdot)$ calculus. More concretely, similar to other proposed methods in this category, the proposed framework models the structure of the machine learning algorithm as an SCM. It then proposes a scalable causal inference approach to the estimate individual treatment effect of a desired component on the decision made by the algorithm.

Chattopadhyay et al. suggest to simplify the SCM defined on a multi-layer network $M([l_1, l_2, l_3...., l_n], U, f, P_U)$ to another network as SCM $M'([l_1, l_n], U, f', P_U)$ where l_1 and l_n represent neurons in the input and output layers, l_i represents neurons in the i-th layer of the network, U denotes the set of unknown variables, f and f' correspond to the SCM functions and P_U defines distributions of the unknown variables. They then propose to calculate the ACE of any neurons of the model on the output by performing causal reasoning on M as follows,

$$ACE_{do(x_i=\alpha)}^y = \mathbb{E}[y|do(x_i = \alpha)] - baseline_{x_i}, \quad (3)$$

where x_i is i-th neuron of the network, y is the output of the model and α is an arbitrary value the neuron is set to. They also propose to calculate the $baseline_{x_i}$ as $\mathbb{E}_{x_i}[\mathbb{E}_y[y|do(x_i = \alpha)]]$

In another research direction, Zhao and Hastie [113] state that to extract the causal interpretations from black-box models, one needs a model with good predictive performance, domain knowledge in the form of a causal graph, and an appropriate visualization tool. They further explore partial dependence plot (PDP) [23] and Individual Conditional Expectation (ICE) [28] to extract causal interpretations from black-box models. Alvarez-Melis and Jaakkola [3] generated causal explanations for structured input structured output black-box models by (a) generating perturbed samples using a variational auroencoder; (b) generating a weighted bipartite graph $G = (V_x \cup V_y, E)$, where V_x and V_y are elements in x and y and E_{ij} represents the causal influence of x_i and y_j; and (c) generating explanation components using graph partitioning algorithms.

Parafita and Vitria [81] introduce a causal attribution framework to explain decisions of a classifier based on the latent factors. The framework consists of three steps, (a)

constructing Distributional Causal Graph which allows us to sample and compute likelihoods of the samples; (b) generating a counterfactual image which is as similar as possible to the original image; and (c) estimating the effect of the modified factor by estimating the causal effect.

Causal interpretation has also gained a lot of attention in Generative Adversarial Networks (GANs) interpretability. Bau et al. [7] propose a causal framework to understand "How" and "Why" images are generated by Deep Convolutional GANs (DCGANs). This is achieved by a two-step framework which finds units, objects or scenes that cause specific classes in the data samples. In the first step, dissection is performed, where classes with explicit representations in the units are obtained by measuring the spatial agreement between individual units of the region we are examining and classes using a dictionary of object classes. In the second step, intervention is performed to estimate the causal effect of a set of units on the class. This framework is then used to find the units with the highest causal effect on the class. Following equation shows the objective of this framework,

$$\alpha^* = \arg\min_\alpha(-\delta_{\alpha \to c} + \lambda||\alpha||_2), \quad (4)$$

where α indicates the units that have causal effect on the outcome, $\delta_{\alpha \to c}$ measures the causal effect of units on the class by intervening on α and set it to the constant c and $\lambda||\alpha||_2$ is a regularization term. Besserve et al. [8] propose to better understand the internal functionality of generative models such as GANs or Variational Autoencoders (VAE) and answer questions like "For a face generator, is there an internal encoding of the eyes, independent of the remaining facial features?", by manipulating the internal variables using counterfactual inference.

Madumal et al. [68] leverage causal inference to explain the behavior of reinforcement learning agents by learning an SCM during reinforcement learning and generate counterfactual examples using the learned SCM.

4.2 Causal Inference and Example-based Interpretation

As mentioned in Section 2.2, in example based explanations, we are looking for data instances that are capable of explaining the model or the underlying distribution of the data. In this subsection, we explain *counterfactual explanations*, a type of example-based explanations, which are one of the widely used explanations for interpreting a model's decisions. Counterfactual explanations aim to answer "Why" questions such as "Why is the model's decision Y?" or "Was it input X that caused the model to predict Y?". Generally speaking, counterfactuals are designed to answer hypothetical questions such as "What would have happened to Y, had I not done X?". They are designed based on a new type of conditional probability $P(y_x|x', y')$. This probability indicates how likely the outcome (label) of an observed instance, i.e., y', would change to y_x if x' is set to x. These kinds of questions can be answered using SCMs [25].

Counterfactual explanations are defined as examples that are obtained by performing minimal changes in the original instance's features and have a predefined output. For example, what minimal changes can be made in a credit card applicant's features such that their application gets accepted. These explanations are human friendly because they are usually focused on a few number of features and there-

fore are more understandable. However, they suffer from the *Roshomon* effect [71] which means there could be multiple true versions of explanations for a predefined outcome. To alleviate this problem, we could report all possible explanations, or find a way to evaluate all explanations and report the best one. Recently, several works have been proposed to generate counterfactual explanations. In order to generate counterfacutal examples, Wachter et al. [101] propose to minimize the mean squared error between the model's predictions and counterfactual outcomes as well as the distance between the original instances and their corresponding counterfactuals in the feature space. Eq. (5) shows the objective function to achieve this goal,

$$\underset{x_{cf}}{\arg\min} \max_{\lambda} L(x, x_{cf}, y, y_{cf})$$
$$L(x, x_{cf}, y, y_{cf}) = \lambda \cdot (\hat{f}(x_{cf}) - y_{cf})^2 + d(x, x_{cf}), \quad (5)$$

where the first term indicates the distance between the model's prediction for the counterfactual input x_{cf} and the desired counterfactual output, while the second term indicates the distance between the actual instance features x and the counterfactual features x_{cf}.

Liu et al. [60] propose a generative model to generate counterfactual explanations for explaining a model's decisions using Eq.(5). Garth et al. [35] propose a method to generate counterfactual examples in a high dimensional setting. The method is proposed for credit application prediction via off-the-shelf interchangeable black-box classifiers. In the case of high dimensional feature space, the generated explanation might not be interpretable due to the existence of too many features. To alleviate the problem, the authors propose to reweigh the distance between the features of an instance and its corresponding counterfactual with the inverse median absolute deviation (Eq.(6)). This metric is robust to outliers and results in more sparse, and therefore, more explainable solutions.

$$MAD_j = median_{i \in \{1,2,...,n\}}(|x_{i,j} - median_{l \in \{1,2,...,n\}}(x_{l,j})|) \quad (6)$$

Goyal et al. [34] propose to generate counterfactual visual explanations for a query image I by using a distractor image I' which belongs to the class c' (a different class from the actual output of the classifier). To generate counterfactual explanations, the authors propose to detect spatial regions in I and I' such that replacing those regions in I with regions in I' results in system classifying the generated image as c'. In order to avoid trivial solutions such as replacing the entire image I with I', authors propose to minimize the number of edits to transform I to I'. The proposed framework is shown in the following equation,

$$\min_{P,a} ||a||_1$$
$$\text{s.t. } c' = \arg\max g((\mathbf{1} - a) \circ f(I) + a \circ Pf(I')) \quad (7)$$
$$a_i \in \{0, 1\} \ \forall i \text{ and } P \in \mathcal{P},$$

where $a \in \mathbb{R}^{hw}$ (h and w represent height and width of an image, respectively) is a binary vector which indicates whether the feature in I needs to be changed with the feature in I' (value 1) or not (value 0). $P \in \mathbb{R}^{hw \times hw}$ is a permutation matrix used to align spatial cells of $f(I')$ with $f(I)$, $f(I)$ and $f(I')$ correspond to spatial feature maps of I and I', respectively. Function $g(.)$ represents the classifier and $\mathcal{P}$ is a set of all $hw \times hw$ permutation matrices. Goyal

et al. [33] propose to explain classifiers' decisions by measuring the *Causal Concept Effect* (CACE). CACE is defined as the causal effect of a concept (such as the brightness or an object in the image) on the prediction. In order to generate counterfactuals, authors leverage a VAE-based architecture. Hendricks et al. [45] propose a method to generate counterfactual explanations using multimodal information for video classification tasks. The proposed method in this work generates visual-linguistic explanations in two steps. First, it trains a classification model for which we would like to generate explanations. Then, in the second step, it trains a post-hoc explanation model by leveraging the output and mid-level features of the trained model in first the step. The explanation model predicts the counterfactuality score for all the negative classes (classes that the instance does not belong to according to the prediction model trained in the first step). The explanation model then generates explanations by maximizing the counterfactuality score between positive and negative classes.

Moore et al. [73] propose to leverage adversarial examples to generate counterfactual explanations. In order to generate plausible explanations, the number of changed features should be small. Moreover, some features such as age cannot be changed arbitrarily. For example, we cannot ask loan applicants to reduce their age. Therefore, to constrain the number of changed features and the direction of gradients in the generated adversarial examples, authors propose to mask the unwanted features and gradients in a way that only desired features change in the generated explanations.

Kommiya et al. [76] propose to explain the decision of a machine learning framework by generating counterfactual examples which satisfy the following two criteria, (1) generated examples must be feasible given user's conditions and context such as range for the features or features to be changed; (2) counterfactual examples generated for explanations should be as diverse as possible. In order to impose the diversity criterion, authors propose to either maximize the point-wise distance between examples in feature-space or leverage the concept from Determinantal point processes to select a subset of samples with the diversity constraint.

Van Looveren and Klaise [61] propose to leverage class prototypes to generate counterfactual explanations. They also claim that using class prototypes for counterfactual example generation accelerates the process. This work suggests that the generated examples by traditional counterfactual generation frameworks [101, 35] do not satisfy two main criteria: (1) they do not consider the training data manifold which may result in out-of-distribution examples, and (2) the hyperparameters in the framework should be carefully tuned in an appropriate range which could be time consuming. To solve the mentioned problems, the authors propose to add a reconstruction loss term (defined as L_2 reocnstruction error between counterfactuals and an autoencoder trained on the training samples) as well as a prototype loss term, which is defined as L_2 loss between the class prototype and the counterfactual samples, to the original objective function of counterfactual generation (Eq 5).

Rathi [86] proposes to generate counterfactual explanations using shapely additive explanations (SHAP).Hendricks et al. [39] defined a method to generate natural language counterfactual explanations. The framework checks for evidences of a counterfactual class in the text explanation generated for the original input. It then checks if those factors

exist in the counterfactual image and returns the existing ones.

4.3 Causal Inference and Fairness

Nowadays, politicians, journalists and researchers are concerned regarding the interpretability of model's decisions and whether they comply with ethical standards [31]. Algorithmic decision making has been widely utilized to perform different tasks such as approving credit lines, filtering job applicants and predicting the risk of recidivism [15]. Prediction of recidivism is used to determine whether to detain or free a person and therefore, it needs to be guaranteed that it does not discriminate against a group of people. Since conventional evaluation metrics such as accuracy does not take these into account, it is usually required to come up with interpretable models in order to satisfy fairness criteria. Recently, huge attention has been paid to incorporating fairness into decision making methods and its connection with causal inference. Kusner et al. [53] propose a new metric for measuring how fair decisions are based on counterfactuals. According to this paper, a decision is fair for an individual if the outcome is the same both in the actual world and a counterfactual world in which the individual belonged to a different demographic group. Kilbertus et al. [46] address the problem from a data generation perspective by going beyond observational data. The authors propose to utilize causal reasoning to address the fairness problem by asking the question "What do we need to assume about the causal data generating process?" instead of "What should be the fairness criterion?".

Madras et al. [67] propose a causal inference model in which the sensitive attribute confounds both the treatment and the outcome. It then leverages deep learning techniques to learn the parameters of the model. Zhang and Bareinboim [110] propose a metric (i.e., causal explanations) to quantitatively measure the fairness of an algorithm. This measure is based on three measures of transmission from cause to effect namely counterfactual direct (Ctf-DE), indirect (Ctf-IE), and spurious (Ctf-SE) effects as defined below. Given an SCM M, the counterfactual indirect effect of intervention $X = x_1$ on $Y = y$ (relative to baseline $X = x_0$) conditioned on $X = x$ with mediator $W = W_{x_1}$ is defined as,

$$IE_{x_0,x_1}(y|x) = P(y_{x_0,W_{x_1}}|x) - P(y_{x_0}|x) \qquad (8)$$

the counterfactual direct effect of intervention $X = x_1$ on Y (with baseline x_0) conditioned on $X = x$ is defined as,

$$DE_{x_0,x_1}(y|x) = P(y_{x_1,W_{x_0}}|x) - P(y_{x_0}|x) \qquad (9)$$

And finally, the spurious effect of event $X = x_1$ on $Y = y$ (relative to baseline x_0) is defined as,

$$SE_{x_0,x_1}(y|x) = P(y_{x_0}|x_1) - P(y|x_0) \qquad (10)$$

Wu et al. [102] unify all causal-based notions for fairness into one general notion called path-specific counterfactual fairness (PC fairness). They then introduce a constraint optimization problem for this notion.

4.4 Causal Inference as Guarantee for Interpretability

Machine learning has had great achievements in medical, legal and economic decision making. Frameworks for these applications must satisfy the following two criteria: 1) they must be causal 2) they must be interpretable. For example, in order to find the efficacy of a drug on patient's health, one needs to estimate the causal effect of the drug on patient's health status. Moreover, in order for the results to be reliable for doctors and experts, an explanation of how the decision has been made is necessary. Despite recent achievements in these two fields separately, not so many works have been done to cover both requirements simultaneously. Moreover, the state-of-the-art approaches in each field are incompatible and therefore can not be combined and used together. Kim and Bastani [49] propose a framework to bridge the gap between causal and interpretable models by transforming *any* algorithm into an interpretable individual treatment effect estimation framework. To be more specific, this work leverages the algorithm proposed in [92] to learn an oracle function f which estimates the causal effect of a treatment for any observed instance and then learn an interpretable function f' to estimate f. They further provide a bound for the error produced by their framework.

In another line of research, causal interpretability has been used to verify the causal relationships in the data. Caruana et al. [11] perform two case studies to discover the rules which show cases where generalized additive models with pairwise interactions (GA^2Ms) learn rules based on only correlations in the data and invade causal rules. They then propose to fix the learned rules based on domain experts knowledge.

Bastani et al. [88] propose a decision tree based explanation method to generate global explanations for a black-box model. Their proposed framework provides powerful insights into the data such as causal issues confirmed by the physicians previously.

5. PERFORMANCE EVALUATION

In this section we provide a detailed review of evaluation methods and common datasets used to assess the interpretability of models for causal interpretablity. Evaluation of interpretability is a challenging task due to the lack of consensus definition of interpretability and understanding of humans from the concept. Evaluation of causal interpretability is even more challenging due to the lack of groundtruth data for causal explanations and verification of causal relationships. Therefore, it is important to have a unified guideline on how to evaluate the proposed models. Traditional interpretability of a model is usually measured with quantifiable proxies such as if a model is approximated using sparse linear models it can be considered interpretable. To evaluate the causal interpretability, researchers also came up with some proxy metrics such as size and diversity of the counterfactual explanation. In this section, we discuss all criteria defined for the "goodness" of both causal and traditional interpretations and proxy metrics to measure how good the proposed framework can generate these explanations.

5.1 Datasets

In this section, we briefly introduce benchmark datasets commonly used to evaluate interpretable models. Depending on the type of the data (i.e., text, image or tabular) different datasets are used to assess the interpretability . Some commonly used datasets for image are "ImageNet (ILSVRC)" [90], "MNIST" [56] and "PASCAL VOC dataset" [21]. While for text they experimented on "20 Newsgroup Dataset" [54],

"Yelp" [108], "IMDB" [43] and "Amazon" [5] reviews. "UCI repository" [97] consists of some tabular datasets that were used by the literature such as "Spambase", "Insurance", "Magic", "Letter", and "Adult" datasets. In order to explain the outcome of the test sample, the explanations are provided by the model. For instance, in the case of image data, those patches of the images that are mostly responsible for the class label were selected. For the text data, words involved in the final decision are made bold with different shades of color, which represent the degree of their involvement. In addition to the mentioned datasets, there are some datasets commonly used to evaluate the causal interpretable frameworks. In the following, we list common datasets used for the evaluation of causal interpretability.

- *German loan dataset* [19]. This dataset contains 1000 observations of loan applicants which contains, numeric, categorical and ordinal attributes.

- *LendingClub*. This dataset[2] contains 5 years of loan records (2007-2011) given by LendingClub company. After preprocessing, it contains 8 features, namely, employment years, annual income, number of open credit accounts, credit history, loan grade as decided by LendingClub, home ownership, purpose, and the state of residence in the United States. .

- *COMPAS*. Collected by ProPublica [22] for analysis purposes on recidivism decisions in the United States, after preprocessing, this data contains 5 features, namely, bail applicant's age, gender, race, prior count of offenses, and degree of criminal charge.

Unfortunately, datasets used for this purpose are not specifically designed for causal interpretability and do not contain the groundtruth that captures the causal aspect of the model such as counterfactual explanations or the ACE of different components of the model on the final decision. On the other hand, there are existing benchmark datasets specifically designed for evaluating tasks in causal inference. Cheng et al. [58] provide a comprehensive survey on benchmark datasets for different causal tasks.

5.2   Evaluation Metrics

In order to assess the performance of a causal interpretable framework, authors are required to evaluate the interpretability of generated explanations from two aspects, (1) the quality of the generated explanations, i.e., are generated explanations interpretable to humans?; and (2) are the generated explanations causal? In the following two subsections, we provide comprehensive guidelines and metrics on how to answer these questions.

5.2.1   Interpretability Evaluation Metrics

Evaluating the interpretability of a machine learning model is usually a challenging task. Interpretable frameworks often evaluate their methods via two main perspectives, (1) how well the generated explanations by the method match the human expectation from different aspects; (2) how well the generated explanations are without using any human subjects. Thus, we will categorize different assessment methods based on the aforementioned perspectives and provide some examples of experiments conducted by the researchers.

Human Subject-Based Evaluation Metrics. Part of the research in interpretability aims to let humans understand the reasons behind the outcome of a product. Accordingly, experiments carried out by the researchers usually answer the following questions:

- By providing two different models, can the explanations help users choose the better classifier in terms of generalizability? This will help us to investigate whether the explanations can be used to decide which model is better. Ribeiro et al. [89] used human subjects from "Amazon Mechanical Turk" (AMT) to choose between two models, one that generalizes better than the other while its accuracy was lower on cross validation. With the provided explanations, the subjects were able to choose the more generalized model 89% of the time.

- With explanations provided by the interpretable methods for a particular sample, can a user correctly predict the outcome of that sample? This is also called "Forward Simulation/Prediction" by Doshi-Velez and Kim [17]. We can verify the explanations actually defines the output we are looking for.

- Based on the explanations, do users trust the classifier to be used in real-world applications? Selvaraju et al. [91] evaluated the trust by asking 54 AMT workers to rate the reliability of the models via a 5-point scale questionnaire. A sample along with its explanations were demonstrated to subjects for two different models, AlexNet and VGG-16 (VGG-16 is known to be more reliable than AlexNet). Moreover, only those instances that provided the same prediction and were aligned with the ground truth label were considered. The results of the evaluation shows that with the proposed explanation the subjects trust the model that generalizes better (VGG-16).

- Do the resulted explanations match human intuition? The model is described to human subjects in detail and they were asked to provide insights about the outcome of the model (human-produced explanations). The test assumes that the explanations provided by the human should be aligned with one that the model provides [66]. Moreover, experts in a specific field (e.g., doctors) can also be used to provide the explanations (e.g., important factors/symptoms) on the task (e.g., recognizing the disease).

- Given two different explanations from different algorithms, which one provides a better quality explanation? This is also known as "Binary Forced Choice" evaluation metric [17]. This test can be used to compare the different explanations from different interpretable models.

Non-human Based Evaluation Metrics. Multiple factors such as human fatigue, improper practice sessions and incentive costs can affect experimental results when human-subject evaluation metrics are used. Hence, it is important to conduct other evaluation metrics.

- How much a proposed interpretable model recovers the important features of the data for a certain prediction task? This requires the important features to be

	Counterfactual Property	Description of Property	Evaluation Metrics
1	Sparsity/Size	Perturbation which transforms x to x_{cf} should be small	Elastic net loss term $(EN(\delta) = \beta.||\delta||_1 + ||\delta||_2^2)$ [61]
			Counting number of altered features manually [35]
2	Interpretability	Counterfactual explanations should lie close to data manifold	Ratio of the reconstruction errors of counterfactual generator trained only on the counterfactual class and counterfactual generator trained on the *original* class [61]
			Ratio of the reconstruction errors of counterfactual generator trained only on the counterfactual class and counterfactual generator trained on the *all* class [61]
3	Proximity	Counterfactual explanations should be as similar as possible to the original instance	$Proximity = -\frac{1}{k} \sum_{i=1}^{k} dist(x_{cf_i}, x)$ [76]
4	Speed	Generating counterfactuals should be fast enough to be deployable in real-world applications	Measure the time and number of gradient updates [61]
5	Diversity	Counterfatual explanations generated for a data instance should be different from each other	$Diversity = \frac{1}{|C_k|^2} \sum_{i=1}^{k-1} \sum_{j=i+1}^{k} dist(x_{cf_i}, x_{cf_j})$ [76]
6	Visual-Linguistic Counterfactuals	Visual explanation is the region which retains high positiveness or negativeness (i.e., on the model prediction for specific positive or negative classes).	Measure how the output of the target classifier changes corresponding to the negative class when a specific region is removed from the input using accuracy [45].
		Linguistic explanation is compatible to the visual counterpart.	Measure how the output of the target classifier changes corresponding to the negative class when a specific region is removed from the input using accuracy [45].

Table 1: A summary of evaluation metrics for counterfactual explanations

known beforehand. We should verify that the model will pick up the important features of the data. One simply can use any base method introduced in Section 2.2.1 as a proxy model to extract the important features. The fraction of these important features recovered by the interpretable method can be used as an evaluation score [89].

- How is locally faithful the proposed method compared to the original model (fidelity)? Lack of fidelity will result in a limited insight to the original model [105]. In convolutional neural network, one common approach is the image occlusion. The pixels that the interpretable method defines as important will be masked to see whether it reflects on the classification score or not [91, 109].

- How consistent the explanations are for the similar instances with the same class label? The explanations should not be significantly different for samples with the same label with a slightly different features. This instability could be the result of a high variance as well as the non-deterministic components of the explanation method [72].

5.2.2 Causal Evaluation Metrics

Due to the lack of groundtruth for causal explanations, to verify the causal aspect of the proposed framework, we need to quantify the desired characteristics of the model and measure the "goodness" of them via some predefined proxy metrics. In the following, we go over the existing metrics to evaluate the proposed causal interpretable frameworks for different categories of causal interpretability.

Counterfactual Explanations Evaluation Metrics. Existing approaches for causal interpretability are mostly based on generating counterfactual explanations. For such approaches, the causal interpretability is often measured through the goodness of generated counterfactual explanation. As mentioned in Section 4, a counterfactual explanation is the highest level of explanation and therefore, we can claim that if an explanation is a counterfactual explanation and is generated by considering causal relationships, it is indeed explainable. However, due to the lack of groundtruth for counterfactuals, we are unable to measure if the generated explanations are generated based on causal relationships. Therefore, to measure the "goodness" of counterfactual explanations, we suggest to conduct experiments to (1) measure the interpretability of the explanations using the metrics designed for interpretability; and (2) evaluate the conterfactuals themselves by measuring different characteristics of them. An interpretable Counterfatual explanation should have the following characteristics:

- The model prediction on the counterfactual sample (x_{cf}) needs to be close to the predefined output for counterfactual explanation.

- The perturbation δ changing the original instance x into $x_{cf} = x + \delta$ should be sparse. In other words, size of counterfactual (i.e., number of features) should be small.

- A counterfactual explanation x_{cf} is considered interpretable if it lies close to the model's training data distribution.

- The counterfactual instance x_{cf} needs to be found fast enough to ensure it can be used in a real life setting.

- Counterfatual explanations generated for a data instance should be different from each other. In other words, counterfactual explanations should be diverse.

- Visual-linguistic counterfactual explanations must satisfy the following two criteria, (1) Visual explanation is the region which keeps high positiveness/negativeness on the model prediction for specific positive/negative classes; (2) Linguistic explanation should be compatible to the visual counterpart in the generated visual explanations.

Below, we briefly discuss these evaluation metrics designed to assess aforementioned characteristics of a counterfactual explanation:

To evaluate the sparsity of the generated counterfactual examples, Mc Grath et al. [35] measures the size of a generated example by counting the number of features each example consists of. Van Looveren and Klaise [61] use elastic net loss term $EN(\delta) = \beta||\delta||_1 + ||\delta||_2^2$ where δ is the difference between the original instance and its generated counterfactual example and β is the hyperparameter.

In order for counterfactual explanations to be interpretable, they need to be close to the data manifold. Looveren and Klaise improves this criterion by suggesting that the counterfactuals are interpretable if they are close to the data manifold of the counterfactual class [61]. To measure the interpretability defined above, Looveren and Klaise propose to measure the ratio of the reconstruction errors when the model used for generating counterfactuals is trained only on the counterfactual class vs when it is trained on the original class [61]. The proposed metric is shown in the following equation,

$$IM1(AE_i, AE_{t_0}, x_{cf}) = \frac{||x_0 + \delta - AE_i(x_0 + \delta)||_2^2}{||x_0 + \delta - AE_{t_0}(x_0 + \delta)||_2^2 + \epsilon} \tag{11}$$

Where AE_i and AE_{t_0} represent the autoencoders used to generate the counterfactuals trained on the class i (counterfactual class) and class t_0 (the original class), respectively. We let x_{cf} and x_0 be the counterfactual explanation and the original sample. In addition, δ denotes the distance between the original and counterfactual samples. A lower value of $IM1$ shows that counterfactual examples can be better reconstructed from the autoencoder trained on the counterfactual class in comparison to the autoencoder trained on the original class. This implies that the generated counterfactuals are closer to the counterfactual class data manifold.

Another metric proposed by [61] measures how similar the generated counterfactuals are when generated using the autoencoder trained on only counterfactuals vs the autoencoder trained on all classes. The metric is shown in the following equation,

$$IM2(AE_i, AE_{t_0}, x_{cf}) = \frac{||AE_i(x_0 + \delta) - AE(x_0 + \delta)||_2^2}{||x_0 + \delta||_1 + \epsilon} \tag{12}$$

A lower value of $IM2$ shows that counterfactuals generated by both autoencoders trained on all classes and counterfactuals are more similar. This implies that the generated counterfactual distribution is as good as the distribution over all classes.

Generated counterfactual explanations can be used to measure users' understanding of a machine learning model's local decision boundary. Mothilal et al. [76] propose to mimic users' understanding of a model's local decision boundaries by, (a) constructing an auxiliary classifier on both original inputs and counterfactual examples; and (b) measuring how well it mimics the actual decision boundaries. More specifically, they train a 1-nearest neighbor (1-NN) classifier on both the original and the counterfactual samples to predict the class of new inputs. The accuracy of this model is then compared with the accuracy of the original model.

The definition of counterfactual explanations implies that generated explanations should be as similar as possible to the original instance. In order to evaluate the proximity between original samples and counterfactual explanations, Mothilal et al. [76] defines proximity as Eq. (13),

$$Proximity = -\frac{1}{k}\sum_{i=1}^{k} dist(x_{cf_i}, x) \tag{13}$$

In order to be able to calculate the proximity for both categorical and continuous features, the authors further propose two metrics to calculate the proximity for categorical and continuous features. For continuous features, the proximity is defined as the mean of feature-wise L_1 distances between the original sample and counterfactuals divided by the median absolute deviation (MAD) of the feature's values in the training set. For categorical features, distance function is calculated such that for each categorical feature it assigns 1 if the feature differs from the original feature and otherwise it assigns 0.

In order to gauge the speed of generating counterfactual explanations, Looveren and Klaise [61] measure the time and the number of gradient updates until the desired counterfactual explanation is generated.

Diversity of generated counterfactuals is measured via measuring feature-wise distances between each pair of counterfactual examples and calculating diversity as the mean of the distances between each pair of examples [76]. Eq. (14) illustrates the measure used for diversity.

$$Diversity = \frac{1}{|C_k|^2}\sum_{i=1}^{k-1}\sum_{j=i+1}^{k} d(x_{cf_i}, x_{cf_j}) \tag{14}$$

Where C_k represents a set of k counterfactuals generated for the original input, x_{cf_i} and x_{cf_j} are the i-th and j-th counterfactuals in the set C_k.

Kanehira et al. [45] propose metrics to evaluate visual-linguistic counterfactual explanations to ensure, (a) visual explanations possess high positiveness/negativeness on the model predictions for positive/negative classes; (b) linguistic explanations are compatible with their corresponding visual explanations. To measure if the generated examples meet these criteria, authors in [45] propose two metrics based on the accuracy. More specifically, to check for the first condition, they investigate how the output of the target classifier changes towards the negative class when a specific region is removed from the input. To measure the second criterion, for each output pair (s, R) they examine how the region R makes the concept s distinguishable by humans. To measure this quantitatively, they compute the accuracy by utilizing bounding boxes for each attribute in the test set. More specifically, IoU (intersection over union) between a given R and all bounding boxes R_0 corresponding to attribute s_0

Overview of interpretable models and their categories			
Traditional Interpretability	Interpretable Models: [107], [104], [64], [103], [65], [106], [50], [13], [40]	Causal Interpretability	Model-based: [77], [38], [12], [8], [113], [81], [7],
			Example-based: [35], [45], [39], [101], [76], [86], [73], [61], [60], [33], [34]
	Post-hoc: [47], [89], [66], [93], [91], [109], [20], [16], [96], [62]		Fairness: [53], [46], [67], [110]
			Guarantee: [49], [17], [88]

Table 2: A summary of the state-of-the-art frameworks for each type of interpretability

is calculated. Then the accuracy is measured by selecting the attribute s_0 with the largest IoU score and checking its consistency with s a counterpart of R.

Table 1 summarizes evaluation metrics for counterfactulas explanations based on the properties of the generated examples.

Model-based Evaluation Metrics. Due to the lack of evaluation groundtruth for representing the actual effect of each component of the model on its final decisions, evaluation for this type of models is still an open problem. One common way of evaluating such models is to report the most important components of a model by measuring their causal effects on the outcome of the model [38, 77]. Chattopadhyay et al. also use the causal attribution of each neuron on the output to visualize the local decisions of the model by saliency map. Moreover, to further investigate how well the model estimates the ACE, they proposed to run the model on datasets for causal effect estimations [12].

Causal Fairness Evaluation. Evaluation of causal fairness models is a challenging task. Papers in this field usually assess the performance of the model for detecting discrimination. Zhang et al. leverage direct, indirect and spurious effect measures (defined in Section 4.3) to detect and explain discrimination [110]. However, to the best of our knowledge, no quantitative measure of causality of a fairness algorithm exists.

6. CONCLUSION

In this survey, we introduce the problem of interpretability in machine learning. We view the problem from two perspectives, (1) traditional interpretability algorithms; (2) causal interpretability algorithms. However, the primary focus of the survey is on causal frameworks. We first provide different definitions of interpretability, then review the state-of-the-art methods in both categories and point out the differences between them. Each type of interpretable models is further subdivided into other sub categories to provide readers with a better overview of existing directions and approaches in the field. More concretely, for traditional methods, we divide existing work into inherently interpretable models and post-hoc interpretability. For causal models, we divide the existing works into the following four categories: counterfactual examples, model-based interpretability, causal models in fairness and interpretability for verifying causal relationships. We also address the challenging problem of evaluating interpretable models, explain existing metrics in detail and categorize them based on the scenarios they are designed for. Table 2 summarizes state-of-the-art methods which belong to each category of interpretability.

ACKNOWLEDGEMENTS

7. REFERENCES

[1] A. Aamodt and E. Plaza. Case-based reasoning: Foundational issues, methodological variations, and system approaches. *AI communications*, 7(1):39–59, 1994.

[2] J. Adebayo, J. Gilmer, M. Muelly, I. Goodfellow, M. Hardt, and B. Kim. Sanity checks for saliency maps. In *Advances in Neural Information Processing Systems*, pages 9505–9515, 2018.

[3] D. Alvarez-Melis and T. Jaakkola. A causal framework for explaining the predictions of black-box sequence-to-sequence models. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 412–421, Copenhagen, Denmark, Sept. 2017. Association for Computational Linguistics.

[4] J. Angwin, J. Larson, L. Kirchner, and S. Mattu. Machine bias risk assessments in criminal sentencing. https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing, Mar 2019.

[5] AWS. Amazon customer reviews dataset. https://s3.amazonaws.com/amazon-reviews-pds/readme.html, 2020.

[6] D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

[7] D. Bau, J. Zhu, H. Strobelt, B. Zhou, J. B. Tenenbaum, W. T. Freeman, and A. Torralba. GAN dissection: Visualizing and understanding generative adversarial networks. *CoRR*, abs/1811.10597, 2018.

[8] M. Besserve, R. Sun, and B. Schölkopf. Counterfactuals uncover the modular structure of deep generative models. *CoRR*, abs/1812.03253, 2018.

[9] D. Boyd and K. Crawford. Critical questions for big data: Provocations for a cultural, technological, and scholarly phenomenon. *Information, communication & society*, 15(5):662–679, 2012.

[10] O. Boz. Extracting decision trees from trained neural networks. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 456–461. ACM, 2002.

[11] R. Caruana, Y. Lou, J. Gehrke, P. Koch, M. Sturm, and N. Elhadad. Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '15, pages 1721–1730, New York, NY, USA, 2015. ACM.

[12] A. Chattopadhyay, P. Manupriya, A. Sarkar, and V. N. Balasubramanian. Neural network attributions:

A causal perspective. *CoRR*, abs/1902.02302, 2019.

[13] X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *Advances in neural information processing systems*, pages 2172–2180, 2016.

[14] W. Cheng, Y. Shen, L. Huang, and Y. Zhu. Incorporating interpretability into latent factor models via fast influence analysis. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 885–893. ACM, 2019.

[15] A. Chouldechova. Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *CoRR*, abs/1703.00056, 2017.

[16] M. Craven and J. W. Shavlik. Extracting tree-structured representations of trained networks. In *Advances in neural information processing systems*, pages 24–30, 1996.

[17] F. Doshi-Velez and B. Kim. Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*, 2017.

[18] M. Du, N. Liu, and X. Hu. Techniques for interpretable machine learning. *arXiv preprint arXiv:1808.00033*, 2018.

[19] D. Dua and C. Graff. UCI machine learning repository, 2017.

[20] D. Erhan, Y. Bengio, A. Courville, and P. Vincent. Visualizing higher-layer features of a deep network. *University of Montreal*, 1341(3):1, 2009.

[21] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88(2):303–338, June 2010.

[22] A. Flores, K. Bechtel, and C. Lowenkamp. False positives, false negatives, and false analyses: A rejoinder to "machine bias: There's software used across the country to predict future criminals. and it's biased against blacks.". *Federal probation*, 80, 09 2016.

[23] J. H. Friedman. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pages 1189–1232, 2001.

[24] N. Frosst and G. Hinton. Distilling a neural network into a soft decision tree. *arXiv preprint arXiv:1711.09784*, 2017.

[25] L. Gerson Neuberg. Causality: models, reasoning, and inference, by judea pearl, cambridge university press, 2000. *Econometric Theory*, 19:675–685, 08 2003.

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

[27] L. H. Gilpin, D. Bau, B. Z. Yuan, A. Bajwa, M. Specter, and L. Kagal. Explaining explanations: An overview of interpretability of machine learning. In *2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA)*, pages 80–89. IEEE, 2018.

[28] A. Goldstein, A. Kapelner, J. Bleich, and E. Pitkin. Peeking inside the black box: Visualizing statistical learning with plots of individual conditional expectation. *Journal of Computational and Graphical Statistics*, 24(1):44–65, 2015.

[29] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, 2016.

[30] I. J. Goodfellow, J. Shlens, and C. Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

[31] B. Goodman and S. Flaxman. Eu regulations on algorithmic decision-making and a "right to explanation", 2016. cite arxiv:1606.08813Comment: presented at 2016 ICML Workshop on Human Interpretability in Machine Learning (WHI 2016), New York, NY.

[32] B. Goodman and S. Flaxman. European union regulations on algorithmic decision-making and a "right to explanation". *AI Magazine*, 38(3):50–57, 2017.

[33] Y. Goyal, U. Shalit, and B. Kim. Explaining classifiers with causal concept effect (cace). *CoRR*, abs/1907.07165, 2019.

[34] Y. Goyal, Z. Wu, J. Ernst, D. Batra, D. Parikh, and S. Lee. Counterfactual visual explanations. *CoRR*, abs/1904.07451, 2019.

[35] R. M. Grath, L. Costabello, C. L. Van, P. Sweeney, F. Kamiab, Z. Shen, and F. Lécué. Interpretable credit application predictions with counterfactual explanations. *CoRR*, abs/1811.05245, 2018.

[36] R. Guo, L. Cheng, J. Li, P. R. Hahn, and H. Liu. A survey of learning causality with data: Problems and methods. *arXiv preprint arXiv:1809.09337*, 2018.

[37] K. S. Gurumoorthy, A. Dhurandhar, G. Cecchi, and C. Aggarwal. Efficient data representation by selecting prototypes with importance weights, 2017.

[38] M. Harradon, J. Druce, and B. E. Ruttenberg. Causal learning and explanation of deep neural networks via autoencoded activations. *CoRR*, abs/1802.00541, 2018.

[39] L. A. Hendricks, R. Hu, T. Darrell, and Z. Akata. Generating counterfactual explanations with natural language. *CoRR*, abs/1806.09809, 2018.

[40] I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. In *International Conference on Learning Representations*, volume 3, 2017.

[41] G. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[42] A. Hyvärinen and E. Oja. Independent component analysis: algorithms and applications. *Neural networks*, 13(4-5):411–430, 2000.

[43] IMDb. Imdb datasets. https://www.imdb.com/interfaces/, 2020.

[44] I. Jolliffe. *Principal component analysis*. Springer, 2011.

[45] A. Kanehira, K. Takemoto, S. Inayoshi, and T. Harada. Multimodal explanations by predicting

counterfactuality in videos. *CoRR*, abs/1812.01263, 2018.

[46] N. Kilbertus, M. Rojas Carulla, G. Parascandolo, M. Hardt, D. Janzing, and B. Schölkopf. Avoiding discrimination through causal reasoning. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 656–666. Curran Associates, Inc., 2017.

[47] B. Kim, R. Khanna, and O. O. Koyejo. Examples are not enough, learn to criticize! criticism for interpretability. In *Advances in Neural Information Processing Systems*, pages 2280–2288, 2016.

[48] B. Kim, O. Koyejo, and R. Khanna. Examples are not enough, learn to criticize! criticism for interpretability. In *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pages 2280–2288, 2016.

[49] C. Kim and O. Bastani. Learning interpretable models with causal guarantees. *CoRR*, abs/1901.08576, 2019.

[50] D. P. Kingma and M. Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[51] P. W. Koh, K.-S. Ang, H. H. Teo, and P. Liang. On the accuracy of influence functions for measuring group effects. *arXiv preprint arXiv:1905.13289*, 2019.

[52] P. W. Koh and P. Liang. Understanding black-box predictions via influence functions. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1885–1894. JMLR. org, 2017.

[53] M. J. Kusner, J. R. Loftus, C. Russell, and R. Silva. Counterfactual fairness. In I. Guyon, U. von Luxburg, S. Bengio, H. M. Wallach, R. Fergus, S. V. N. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, pages 4069–4079, 2017.

[54] K. Lang. 20 newsgroups. `http://qwone.com/~jason/20Newsgroups/`, 2008.

[55] Q. V. Le. Building high-level features using large scale unsupervised learning. In *2013 IEEE international conference on acoustics, speech and signal processing*, pages 8595–8598. IEEE, 2013.

[56] Y. LeCun, C. Cortes, and C. Burges. The mnist database. `http://yann.lecun.com/exdb/mnist/`, Jan 2020.

[57] J. Li, K. Cheng, S. Wang, F. Morstatter, R. P. Trevino, J. Tang, and H. Liu. Feature selection: A data perspective. *ACM Computing Surveys (CSUR)*, 50(6):94, 2018.

[58] Y. Li, R. Guo, W. Wang, and H. Liu. Causal learning in question quality improvement. In *2019 BenchCouncil International Symposium on Benchmarking, Measuring and Optimizing (Bench'19)*, 2019.

[59] Z. C. Lipton. The mythos of model interpretability. *CoRR*, abs/1606.03490, 2016.

[60] S. Liu, B. Kailkhura, D. Loveland, and Y. Han.

Generative counterfactual introspection for explainable deep learning. *CoRR*, abs/1907.03077, 2019.

[61] A. V. Looveren and J. Klaise. Interpretable counterfactual explanations guided by prototypes. *CoRR*, abs/1907.02584, 2019.

[62] Y. Lou, R. Caruana, and J. Gehrke. Intelligible models for classification and regression. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 150–158. ACM, 2012.

[63] Y. Lou, R. Caruana, J. Gehrke, and G. Hooker. Accurate intelligible models with pairwise interactions. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '13, pages 623–631, New York, NY, USA, 2013. ACM.

[64] J. Lu, C. Xiong, D. Parikh, and R. Socher. Knowing when to look: Adaptive attention via a visual sentinel for image captioning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 375–383, 2017.

[65] J. Lu, J. Yang, D. Batra, and D. Parikh. Hierarchical question-image co-attention for visual question answering. In *Advances In Neural Information Processing Systems*, pages 289–297, 2016.

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

[67] D. Madras, E. Creager, T. Pitassi, and R. S. Zemel. Fairness through causal awareness: Learning latent-variable models for biased data. *CoRR*, abs/1809.02519, 2018.

[68] P. Madumal, T. Miller, L. Sonenberg, and F. Vetere. Explainable reinforcement learning through a causal lens. *CoRR*, abs/1905.10958, 2019.

[69] N. Mehrabi, F. Morstatter, N. Saxena, K. Lerman, and A. Galstyan. A survey on bias and fairness in machine learning. *arXiv preprint arXiv:1908.09635*, 2019.

[70] T. Miller. Explanation in artificial intelligence: Insights from the social sciences. *CoRR*, abs/1706.07269, 2017.

[71] C. Molnar. *Interpretable Machine Learning*. Lulu.com, 2019. `https://christophm.github.io/interpretable-ml-book/`.

[72] C. Molnar. *Interpretable machine learning*. Lulu. com, 2019.

[73] J. Moore, N. Hammerla, and C. Watkins. Explaining deep learning models with constrained adversarial examples. *CoRR*, abs/1906.10671, 2019.

[74] S. Moosavi-Dezfooli, A. Fawzi, and P. Frossard. Deepfool: a simple and accurate method to fool deep neural networks. *CoRR*, abs/1511.04599, 2015.

[75] A. Mordvintsev, C. Olah, and M. Tyka. Inceptionism: Going deeper into neural networks, 2015.

[76] R. K. Mothilal, A. Sharma, and C. Tan. Explaining machine learning classifiers through diverse counterfactual explanations. *CoRR*, abs/1905.07697, 2019.

[77] T. Narendra, A. Sankaran, D. Vijaykeerthy, and S. Mani. Explaining deep learning models using causal inference. *CoRR*, abs/1811.04376, 2018.

[78] C. Olah, A. Mordvintsev, and L. Schubert. Feature visualization. *Distill*, 2017. https://distill.pub/2017/feature-visualization.

[79] C. Olah, A. Satyanarayan, I. Johnson, S. Carter, L. Schubert, K. Ye, and A. Mordvintsev. The building blocks of interpretability. *Distill*, 2018. https://distill.pub/2018/building-blocks.

[80] N. Papernot, P. D. McDaniel, S. Jha, M. Fredrikson, Z. B. Celik, and A. Swami. The limitations of deep learning in adversarial settings. *CoRR*, abs/1511.07528, 2015.

[81] Á. Parafita and J. Vitrià. Explaining visual models by causal attribution. *arXiv preprint arXiv:1909.08891*, 2019.

[82] J. Pearl. *Causality*. Cambridge university press, 2009.

[83] J. Pearl. Theoretical impediments to machine learning with seven sparks from the causal revolution. *CoRR*, abs/1801.04016, 2018.

[84] J. Pearl. The seven tools of causal inference, with reflections on machine learning. *Commun. ACM*, 62(3):54–60, Feb. 2019.

[85] G. Plumb, D. Molitor, and A. S. Talwalkar. Model agnostic supervised local explanations. In *Advances in Neural Information Processing Systems*, pages 2515–2524, 2018.

[86] S. Rathi. Generating counterfactual and contrastive explanations using SHAP. *CoRR*, abs/1906.09293, 2019.

[87] A. Renkl. Toward an instructionally oriented theory of example-based learning. *Cognitive science*, 38(1):1–37, 2014.

[88] M. T. Ribeiro, S. Singh, and C. Guestrin. Model-agnostic interpretability of machine learning. *arXiv preprint arXiv:1606.05386*, 2016.

[89] M. T. Ribeiro, S. Singh, and C. Guestrin. Why should i trust you?: Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144. ACM, 2016.

[90] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015.

[91] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 618–626, 2017.

[92] U. Shalit, F. D. Johansson, and D. Sontag. Estimating individual treatment effect: generalization bounds and algorithms. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 3076–3085. JMLR. org, 2017.

[93] K. Simonyan, A. Vedaldi, and A. Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013.

[94] J. T. Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller. Striving for simplicity: The all convolutional net. *arXiv preprint arXiv:1412.6806*, 2014.

[95] P.-N. Tan. *Introduction to data mining*. Pearson Education India, 2018.

[96] G. G. Towell and J. W. Shavlik. Extracting refined rules from knowledge-based neural networks. *Machine learning*, 13(1):71–101, 1993.

[97] UCI. Uci machine learning repository. https://archive.ics.uci.edu/ml/index.php, 2020.

[98] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[99] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio. Graph attention networks. *arXiv preprint arXiv:1710.10903*, 2017.

[100] U. Von Luxburg. A tutorial on spectral clustering. *Statistics and computing*, 17(4):395–416, 2007.

[101] S. Wachter, B. D. Mittelstadt, and C. Russell. Counterfactual explanations without opening the black box: Automated decisions and the GDPR. *CoRR*, abs/1711.00399, 2017.

[102] Y. Wu, L. Zhang, X. Wu, and H. Tong. Pc-fairness: A unified framework for measuring causality-based fairness. In *Advances in Neural Information Processing Systems*, pages 3399–3409, 2019.

[103] H. Xu and K. Saenko. Ask, attend and answer: Exploring question-guided spatial attention for visual question answering. In *European Conference on Computer Vision*, pages 451–466. Springer, 2016.

[104] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, pages 2048–2057, 2015.

[105] F. Yang, M. Du, and X. Hu. Evaluating explanation without ground truth in interpretable machine learning, 2019.

[106] Z. Yang, X. He, J. Gao, L. Deng, and A. Smola. Stacked attention networks for image question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 21–29, 2016.

[107] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, and E. Hovy. Hierarchical attention networks for document classification. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies*, pages 1480–1489, 2016.

[108] YELP. Yelp dataset. https://www.yelp.com/dataset, 2020.

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

[110] J. Zhang and E. Bareinboim. Fairness in decision-making—the causal explanation formula. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[111] Q. Zhang, Y. Yang, H. Ma, and Y. N. Wu. Interpreting cnns via decision trees. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6261–6270, 2019.

[112] Q.-s. Zhang and S.-C. Zhu. Visual interpretability for deep learning: a survey. *Frontiers of Information Technology & Electronic Engineering*, 19(1):27–39, 2018.

[113] Q. Zhao and T. Hastie. Causal interpretations of black-box models. *Journal of Business & Economic Statistics*, pages 1–10, 2019.