Trustworthy AI: A Computational Perspective

HAOCHEN LIU∗, Michigan State University, USA
YIQI WANG∗, Michigan State University, USA
WENQI FAN, The Hong Kong Polytechnic University, Hong Kong
XIAORUI LIU, Michigan State University, USA
YAXIN LI, Michigan State University, USA
SHAILI JAIN, Twitter, USA
YUNHAO LIU, Tsinghua University, China
ANIL K. JAIN, Michigan State University, USA
JILIANG TANG, Michigan State University, USA

In the past few decades, artificial intelligence (AI) technology has experienced swift developments, changing everyone’s daily life and profoundly altering the course of human society. The intention behind developing AI was and is to benefit humans by reducing labor, increasing everyday conveniences, and promoting social good. However, recent research and AI applications indicate that AI can cause unintentional harm to humans by, for example, making unreliable decisions in safety-critical scenarios or undermining fairness by inadvertently discriminating against a group or groups. Consequently, trustworthy AI has recently garnered increased attention regarding the need to avoid the adverse effects that AI could bring to people, so people can fully trust and live in harmony with AI technologies.

A tremendous amount of research on trustworthy AI has been conducted and witnessed in recent years. In this survey, we present a comprehensive appraisal of trustworthy AI from a computational perspective to help readers understand the latest technologies for achieving trustworthy AI. Trustworthy AI is a large and complex subject, involving various dimensions. In this work, we focus on six of the most crucial dimensions in achieving trustworthy AI: (i) Safety & Robustness, (ii) Nondiscrimination & Fairness, (iii) Explainability, (iv) Privacy, (v) Accountability & Auditability, and (vi) Environmental Well-being. For each dimension, we review the recent related technologies according to a taxonomy and summarize their applications in real-world systems. We also discuss the accordant and conflicting interactions among different dimensions and discuss potential aspects for trustworthy AI to investigate in the future.

CCS Concepts: • Computing methodologies → Artificial intelligence; • General and reference → Surveys and overviews; • Security and privacy;

Additional Key Words and Phrases: artificial intelligence, robustness, fairness, explainability, privacy, accountability, environmental well-being

∗Both authors contributed equally to this work.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

© 2018 Association for Computing Machinery.
Manuscript submitted to ACM
1 INTRODUCTION

Artificial intelligence (AI), a science that studies and develops the theory, methodology, technology, and application systems for simulating, extending, and expanding human intelligence, has brought revolutionary impact to modern human society. From a micro view, AI plays an irreplaceable role in many aspects of our lives. Modern life is filled with interactions with AI applications: from unlocking a cell phone with a face ID, talking to a voice assistant to buying products recommended by e-commerce platforms; from a macro view, AI creates great economic outcomes. The Future of Jobs Report 2020 from the World Economic Forum [144] predicts that AI will create 58 million new jobs in five years. By 2030, AI is expected to produce extra economic profits of 13 trillion U.S. dollars, which contribute 1.2% annual growth to the GDP of the whole world [58]. However, along with their rapid and impressive development, AI systems have also exposed their untrustworthy sides. For example, safety-critical AI systems are shown to be vulnerable to adversarial attacks. Deep image recognition systems in autonomous vehicles could fail to distinguish road signs modified by malicious attackers [364], posing a great threat to passenger safety. In addition, AI algorithms can cause bias and unfairness. Online AI chatbots could produce indecent, racist, and sexist content [354] that offends users and has a negative social impact. Moreover, AI systems carry the risk of disclosing user privacy and business secrets. Hackers can take advantage of the feature vectors produced by an AI model to reconstruct private input data, such as fingerprints [26], thereby leaking a user’s sensitive information. These vulnerabilities can make existing AI systems unusable and can cause severe economic and security consequences. Concerns around trustworthiness have become a huge obstacle to the boosting of AI usage and increasing of economic value. Hence, how to build trustworthy AI systems has become a focal topic in both academia and industry.

In recent years, a large body of literature on trustworthy AI has emerged. With the increasing demand for building trustworthy AI, it is imperative to summarize existing achievements and discuss possible directions for future research. In this survey, we provide a comprehensive overview of trustworthy AI to help newcomers attain a basic understanding of what makes an AI system trustworthy and to help veterans track the latest progress in the field. We clarify the definition of trustworthy AI and introduce six key dimensions of it. For each dimension, we present its concepts and taxonomies and review representative algorithms. We also introduce possible interactions among different dimensions and discuss other potential issues around trustworthy AI that have not yet drawn sufficient attention. In addition to definitions and concepts, our survey focuses on the specific computational solutions for realizing each dimension of trustworthy AI. This perspective makes it distinct from some extant related works, such as a government guideline [325], which suggests how to build a trustworthy AI system in the form of laws and regulations, or reviews [55, 337], which discuss the realization of trustworthy AI from a high-level, non-technical perspective.

According to a recent ethics guideline for AI provided by the European Union (EU) [325], a trustworthy AI system should meet four ethical principles: respect for human autonomy, prevention of harm, fairness, and explicability. Based on these four principles, AI researchers, practitioners, and governments propose various specific dimensions for trustworthy AI [55, 325, 337]. In this survey, we focus on six important and concerning dimensions that have been extensively studied. As shown in Figure 1, they are Safety & Robustness, Non-discrimination & Fairness, Explainability, Privacy, Auditability & Accountability, and Environmental Well-Being. These dimensions are
very vital in real applications. Take the face recognition system as an example, it has been widely adopted in identity authentication. Thus, the face recognition system is desired to make highly robust and accurate predictions under any potential risks and attacks. Meanwhile, human face information is very private and crucial biometric information. It should be carefully protected. Furthermore, it is important to eliminate demographic disparities existent in the performance of face recognition, so that the system shows no bias towards any certain ethnic groups or genders. Also, for the reliability of the face recognition system, it is preferred that the recognition result can be explained in a reasonable way and the system can be audited periodically. In addition, the training process of large-scale face recognition models and the wide deployment of them require huge energy consumption, which may lead to a great amount of carbon emission. Thus, how to ensure the environmental friendliness of face recognition systems is important.

The remaining survey is organized as follows. In section 2, we articulate the definition of trustworthy AI and provide various definitions of it to help readers understand how a trustworthy AI system is defined by researchers from such different disciplines as computer science, sociology, law, and business. We then distinguish the concept of trustworthy AI from several related concepts, such as ethical AI and responsible AI.

In section 3, we detail the dimension of Safety & Robustness, which requires an AI system to be robust to the tiny perturbations of inputs and to be able to make secure decisions. In recent years, numerous studies have shown that AI systems, especially those that adopt deep learning models, can be very sensitive to intentional or unintentional inputs perturbations, posing huge risks to safety-critical applications. For example, as described before, autonomous vehicles can be fooled by altered road signs. Additionally, spam detection models can be fooled by emails with well-designed text [32]. Thus, spam senders can take advantage of this weakness to make their emails immune to the detection system, which would cause a bad user experience.

It has been demonstrated that AI algorithms can learn human discrimination through provided training examples and make unfair decisions. For example, some face recognition algorithms have difficulty detecting faces of African Americans [297] or misclassifying them as gorillas [178]. Moreover, voice dictation software typically performs better
at recognizing a voice from a male than that from a female [294]. In section 4, we introduce the dimension of Non-discrimination & Fairness in which an AI system is expected to avoid unfair bias toward certain groups or individuals.

In section 5, we discuss the dimension of Explainability, which suggests that the AI’s decision mechanism system should be able to be explained to stakeholders (who should be able to understand the explanation). For example, AI techniques have been used for disease diagnosis based on the symptoms and physical features of a patient [306]. In such cases, a black-box decision is not acceptable. The inference process should be transparent to doctors and patients to ensure that the diagnosis is exact in every detail.

Researchers have found that some AI algorithms can store and expose users’ personal information. For example, dialogue models trained on the human conversation corpus can remember sensitive information, like credit card numbers, which can be elicited by interacting with the model [173]. In section 6, we present the dimension of Privacy, which requires an AI system to avoid leaking any private information.

In section 7, we describe the dimension of Auditability & Accountability, which expects that an AI system is assessed by a third party and, when necessary, assign responsibility for an AI failure, especially in critical applications [325].

Recently, the environmental impacts of AI systems have drawn people’s attention, since some large AI systems consume great amounts of energy. As a mainstream AI technology, deep learning is moving toward pursuing larger models and more parameters. Accordingly, more storage and computational resources are consumed. A study [330] shows that training a BERT model [116] costs a carbon emission of around 1,400 pounds of carbon dioxide, which is comparable to that of a round trip trans-America flight. Therefore, an AI system should be sustainable and environmentally friendly. In section 8, we review the dimension of Environmental Well-Being.

In section 9, we discuss the interactions among the different dimensions. Recent studies have demonstrated that there are accordance and conflicts among different dimensions of trustworthy AI [325, 352]. For example, the robustness and explainability of deep neural networks are tightly connected and robust models tend to be more interpretable [128, 342] and vice versa [269]. Moreover, it is shown that in some cases, a trade-off exists between robustness and privacy. For instance, adversarial defense approaches can make a model more vulnerable to membership inference attacks, which increases the risk of training data leakage [326].

In addition to the aforementioned six dimensions, there are more dimensions of trustworthy AI, such as human agency and oversight, creditability, etc. Although these additional dimensions are as important as the six dimensions considered in this article, they are in earlier stages of development with limited literature, especially for computational methods. Thus, in section 10, we discuss these dimensions of trustworthy AI as future directions needing dedicated research efforts.

2 CONCEPTS AND DEFINITIONS

The word “trustworthy” is noted to mean “worthy of trust of confidence; reliable, dependable” in the Oxford English Dictionary or “able to be trusted” in the Dictionary of Cambridge. “Trustworthy” descends from the word trust, which is described as the “firm belief in the reliability, truth, or ability of someone or something” in the Oxford English Dictionary or the “belief that you can depend on someone or something” in the Dictionary of Cambridge. Broadly speaking, trust is a widespread notion in human society, which lays the important foundation for the sustainable development of human civilization. Strictly speaking, some potential risks always exist in our external environment because we cannot completely control people and other entities [248, 337]. It is our trust in these parties that allows us to put ourselves at potential risk to continue interacting with them willingly [219]. Trust is necessary among people. It forms the basis of a
good relationship and is necessary for people to live happily and to work efficiently together. In addition, trust is also vital between humans and technology. Without trust, humans would not be willing to utilize technology, which would undoubtedly impede its advancement and prevent people from enjoying the conveniences it brings. Therefore, for a win-win situation between humans and technology, it is necessary to guarantee that the technology is trustworthy so people can build trust in it.

The term “artificial intelligence” got its name from a workshop at a 1956 Dartmouth conference [57, 249]. Although there are numerous definitions for AI [210], AI generally denotes a program or system that is able to cope with a real-world problem with human-like reasoning, for example, in the field of image recognition within AI, which uses deep learning networks to recognize objects or people within images [288]. The past few decades have witnessed rapid and impressive development of AI; there are tons of breakthroughs happening in every corner of this field [301, 332]. Furthermore, with the rapid development of big data and computational resources, AI has been broadly applied to many aspects of human life, including economics, healthcare, education, transportation, and so on, where it has revolutionized industries and achieved numerous feats. Considering the important role AI plays in modern society, it is necessary to make AI trustworthy so that humans can rely on it with minimal concern regarding its potential harm. Trust is essential in allowing the potential of AI to be fully realized – and humans to fully enjoy its benefits and convenience [98].

Table 1. A summary of principles for Trustworthy AI from different perspectives.

<table>
<thead>
<tr>
<th>Perspective</th>
<th>Principles</th>
</tr>
</thead>
<tbody>
<tr>
<td>Technical</td>
<td>Accuracy, Robustness,</td>
</tr>
<tr>
<td></td>
<td>Explainability</td>
</tr>
<tr>
<td>User</td>
<td>Availability, Usability,</td>
</tr>
<tr>
<td></td>
<td>Safety, Privacy, Autonomy</td>
</tr>
<tr>
<td>Social</td>
<td>Law-abiding, Ethical, Fair,</td>
</tr>
<tr>
<td></td>
<td>Accountable, Environmental-friendly</td>
</tr>
</tbody>
</table>

Due to its importance and necessity, trustworthy AI has drawn increasing attention, and there are numerous discussions and debates over its definition and extension [98]. In this survey, we define trustworthy AI as programs and systems built to solve problems like a human, which bring benefits and convenience to people with no threat or risk of harm. We further define trustworthy AI from the following three perspectives: the technical perspective, the user perspective, and the social perspective. An overall description of these perspectives is summarized in Table 1.

- **From a technical perspective**, trustworthy AI is expected to show the properties of accuracy, robustness, and explainability. Specifically, AI programs or systems should generate accurate output consistent with the ground truth as much as possible. This is also the first and most basic motivation for building them. Additionally, AI programs or systems should be robust to changes so that perturbations would not affect the model outcome. This is very important, since real environments where AI systems are deployed are usually very complex and volatile. Last, but not least, trustworthy AI must allow for explanation and analysis by humans, so that potential risks and harm can be minimized. In addition, trustworthy AI should be transparent so people can better understand its mechanism.

- **From a user’s perspective**, trustworthy AI should possess the properties of availability, usability, safety, privacy, and autonomy. Specifically, AI programs or systems should be available for people whenever they need them, and these programs or systems should be easy to use for people with different backgrounds. More importantly, AI programs or systems are expected to avoid harm under any conditions, and to always put the safety of users
as the first priority. In addition, trustworthy AI should protect the privacy of all users. It should deal with data storage very carefully and seriously. Last but not least, the autonomy of trustworthy AI should always be under people’s control. In other words, it is always a human’s right to grant an AI system any decision-making power or to withdraw that power at any time.

• **From a social perspective**, trustworthy AI should be law-abiding, ethical, fair, accountable, and environmentally friendly. Specifically, AI programs or systems should operate in full compliance with all relevant laws and regulations and comply with the ethical principles of human society. Importantly, trustworthy AI should show nondiscrimination toward people from various backgrounds. It should guarantee justice and fairness among all users. Also, trustworthy AI should be accountable, which means it is clear who is responsible for each part of the AI system. Lastly, for the sustainable development and long-term prosperity of our civilization, AI programs and systems should be environmentally friendly. For example, they should limit energy consumption and cause minimal pollution.

Note that the above properties of the three perspectives are not independent of each other. Instead, they complement and reinforce each other.

There have been numerous terminologies related to AI proposed recently, including ethical AI, beneficial AI, responsible AI, explainable AI, fair AI, and so on. These terminologies share some overlap and distinction with trustworthy AI in terms of the intention and extension of the concept.

Next, we briefly describe some related terminologies to help readers enhance the understanding of trustworthy AI.

• **Ethical AI** [143]: An ethical framework of AI that specifies five core principles, including beneficence, nonmaleficence, autonomy, justice, and explicability. Additionally, 20 specific action points from four categories have been proposed to ensure continuous and effective efforts. They are assessment, development, incentivization, and support.

• **Beneficial AI** [271]: AI has undoubtedly brought people countless benefits, but to gain sustainable benefits from AI, 25 principles have been proposed in conjunction with the 2017 Asilomar conference. These principles are based on three aspects: research issues, ethics and values, and longer-term issues.

• **Responsible AI** [2, 4]: A framework for the development of responsible AI consists of 10 ethical principles: well-being, respect for autonomy, privacy and intimacy, solidarity, democratic participation, equity, diversity inclusion, prudence, responsibility, and sustainable development. The Chinese National Governance Committee for the New Generation Artificial Intelligence has proposed a set of governance principles to promote the healthy and sustainable development of responsible AI. Eight principles have been listed as follows: harmony and human-friendliness, fairness and justice, inclusion and sharing, respect for privacy, safety and controllability, shared responsibility, open and collaborative, and agile governance.

• **Explainable AI** [18]: The basic aim of explainable AI is to open up the “black box” of AI, to offer a trustworthy explanation of AI to users. It also aims to propose more explainable AI models, which can provide promising model performance and can be explained in non-technical terms at the same time, so that users can fully trust them and take full advantage of them.

• **Fair AI** [403]: Because AI is designed by humans and data plays a key role in most AI models, it is easy for AI to inherit some bias from its creators or input data. Without proper guidance and regulations, AI could be biased and unfair toward a certain group or groups of people. Fair AI denotes AI that shows no discrimination toward
people from any group. Its output should have little correlation with the traits of individuals, such as gender and ethnicity.

Overall, trustworthy AI has a very rich connotation and can be interpreted from several perspectives. It contains the concepts of many existing terminologies, including fair AI [403], explainable AI [18], and so on. Huge overlaps also exist among the concept of trustworthy AI and the concepts of ethical AI [143], beneficial AI [271], and responsible AI [2, 4]. Although proposed by different organizations, all of them aim at building reliable AI that sustainably benefits human society and these concepts can be exchangeable in some contexts. Note that Ethical AI [143] was proposed based on six documents including those where Beneficial AI [271] and Responsible AI [2] were proposed. Nonetheless, some differences exist among these concepts. For example, in beneficial AI and responsible AI, there are some principles and requirements for governments around the world, such as to avoid any arms race, but other concepts do not involve them.

3 SAFETY & ROBUSTNESS
A human can trust AI systems in safety-critical scenarios only if the machine learning (ML) systems can achieve stable and sustained high accuracy under the worst circumstances without any vulnerability. Otherwise, ML systems will cause severe consequences in safety-critical scenarios. To be specific, payment apps need to verify user identities through the face recognition system, but the identity theft may create different face images to fool the system and commit the crime [84]; ML systems are also widely used in developing autonomous driving techniques. Dangers may happen when the traffic sign recognition process generates a wrong prediction on the stop sign being slightly perturbed by some white mark [130]. In social networks, someone may pretend to be an important user by connecting with selected people to fool the social media mining algorithms. To avoid such dangerous behaviors, the ML systems should be robust to small perturbations since real-world data contains diverse types of noise. In recent years, many studies have shown that ML models can be fooled by imperceptible perturbations, namely, adversarial perturbations [243, 333]. From traditional ML classifiers [40] to deep learning models, like CNN [333], GNN [311], or RNN [381], none of the models is sufficiently robust to such perturbations. This raises huge concerns when ML models are applied to safety-critical tasks, such as authentication [84], autonomous driving [324], recommendation [132, 137], AI healthcare [142], etc. To build safe and reliable ML models, studying adversarial examples and the underlying reasons is urgent and essential.

In this section, we aim to introduce the concept of robustness, including how to design threat models and different types of defense strategies. We first introduce the concept of adversarial robustness. Then, we provide more details by introducing the taxonomy as well as examples for each category. We then discuss different adversarial attacks and defense strategies by introducing some representative methods. Next, we introduce how adversarial robustness issues affect real-world AI systems. We also present related tools and surveys to help readers further explore this field. Finally, we demonstrate potential future directions in adversarial robustness.

3.1 Concepts and Taxonomy
In this subsection, we describe the common and fundamental concepts and taxonomy in AI robustness to illustrate an overview of adversarial attacks and defenses.

3.1.1 Threat Models. An adversarial threat model is an adversarial attacker that tries to break the performance of ML models with fake training or test examples. The existence of adversarial attacks could lead to serious security concerns in a wide range of ML applications. Attackers use many different types of strategies to achieve their goals.
Therefore, threat models can be categorized into different types. In this subsection, we introduce different categories of threat models aiming to attack the ML system from different perspectives, including when the attack happens, what information of the model the attacker can access, and what the adversary’s goal is.

- **Poisoning Attacks vs. Evasion Attacks.** Whether an attack is evasion or poisoning depends on whether attackers modify the training or test samples. A poisoning attack occurs when attackers add fake samples into the training set of a ML model. These fake samples are designed intentionally to train a bad model, thus the model would achieve bad overall performance [41] or give wrong predictions on a certain group of test samples [404]. When an adversary has access to the training data, this type of attack can happen and will raise realistic safety concerns. For example, the training data for an online shopping recommendation system is often collected from web users where attackers may exist. A special case of poisoning attacks is backdoor attack, which aims to only mislead the performance of test samples with a special trigger known by the attacker [84]. Recently, the connection between poisoning attack and shortcut learning [149, 174, 376] has been noticed. Shortcut learning summarizes a general phenomenon that machine learning models tend to link spurious features with the predictions in the training data while they could not generalize well on the test set. This aligns well with the underlying reason why poisoning attack leads to bad test behaviors.

An evasion attack happens in the test phase. Given a well-trained classifier, attackers aim to design small perturbations for test samples in order to elicit wrong predictions from a victim model. From Figure 2, we can see that the image of a panda can be correctly classified by the model, while the perturbed version will be classified as a gibbon.

- **White-box attacks vs. Black-box attacks.** According to the adversary’s knowledge, attacking methods can be categorized into white-box and black-box attacks. White-Box attacks refer to a setting in which the adversary can utilize all the information of the target model, including the architecture, parameters, gradient information, etc. Generally speaking, the attacking process can be formulated as an optimization problem to optimize the risk of wrong predictions [69, 160]. With the ability to access white-box information, this problem is often much easier to be solved via gradient-based methods. White-box attacks have been extensively studied because the disclosure of model architecture and parameters helps people understand the weakness of ML models clearly; thus, they can be analyzed mathematically.

In the black-Box attack setting, no knowledge of ML models is available to the adversaries. Adversaries can only feed the input data and query the outputs of the models. One of the most common ways to perform black-box attacks is to keep querying the victim model and approximate the gradient through numerical differentiation methods. Compared to white-box attacks, black-box attacks are more practical because ML models are less likely to be white-box due to privacy issues in reality.
• **Targeted Attacks vs. Non-Targeted Attacks.** In the image classification problem, threat models can be categorized by whether the adversary wants to get a pre-set label for a certain image. In **targeted attacks**, a specified target prediction label is expected for each adversarial example in the test phase. For example, identity theft may want to fool the face recognition system and pretend to be a specific important user. In contrast, **non-targeted attacks** expect an arbitrary prediction label except for the real one. Note that in other data domains, like graph data, the definition of targeted attack can be extended to mislead certain groups of nodes, but not necessary to force the model to give a particular prediction for one node.

3.1.2 **Victim Models.** Victim models are the models that are attacked by the attacker. The victim model ranges from traditional machine learning models like SVM [41] to Deep Neural Networks (DNNs), including the Convolution Neural Network (CNN) [217], Graph Neural Network (GNN) [311], Recurrent Neural Network (RNN) [381], etc. In this section, we briefly introduce the victim models that have been studied and shown to be vulnerable to adversarial attacks.

• **Traditional machine learning models.** One of the earliest works about robustness checked the security of Naive Bayes classifiers [109]. Later, SVM and the naive fully-connected neural networks have been shown to be vulnerable to attacks [40]. Recently, the adversarial robustness of tree-based models has also been proposed as an open problem [78].

• **Deep learning models.** In computer vision tasks, Convolution Neural Networks (CNNs) [212] are one of the most widely used models for image classification problems. CNN models aggregate local features from images to learn the representations of image objects and give predictions based on learned representations. The vulnerability of deep neural networks to attack was first proposed in CNN [333]. Since then, there have been extensive works indicating that CNNs are not robust against adversarial attacks. As another type of deep learning model, Graph Neural Networks (GNNs) have been developed for graph-structured data and can be used by many real-world systems, such as social networks and natural science. People also pay effort into testing GNN’s robustness [45, 81, 108, 242, 406] and building robust GNNs [190]. Consider the node classification problem as an example: existing works show that the performance can be reduced significantly by slightly modifying node features, adding or deleting edges, or adding fake nodes [404]. For sequence models that deal with text data, there exist studies focusing on evaluating their robustness. In this scenario, attackers need to consider semantic or phonetic similarity to guarantee unnoticeable perturbation. Therefore, in addition to the commonly used optimization method used to attack a seq2seq translation model [86], some heuristic approaches are proposed to find substitute words that attack RNN-based dialogue generation models [268].

3.1.3 **Defense Strategies.** Under evasion adversarial attacks, there are different types of countermeasures to prevent the adversary from creating harmful effects. During the training stage, **adversarial training** aims to train a robust model by using adversarial samples during the training process. **Certified defense** works to achieve robustness over all perturbations within a certain bound. For defenses that happen at inference time, **adversarial example detection** aims to distinguish adversarial examples, so users can reject the prediction of harmful examples. To defend against attacks happening during training, feature extractors trained on auxiliary data have been used to extract useful features and correspondingly avoid shortcuts that can help prevent poisoning attack [376]. **Adversarial training** has also been shown to be effective against poisoning attacks [181, 335].
3.2 Representative Attack Methods

In this subsection, we introduce representative attack methods from two aspects: evasion attacks and poisoning attacks.

3.2.1 Evasion Attack. Evasion attacks occur at test time. When attackers want to generate an adversarial example for a test example, they need to find a distance measure to guarantee the perturbation size is small so that evasion attacks can be further categorized by how to constraint the perturbation, i.e., pixel constrained adversarial examples with a fixed \( l_p \) norm bound and adversarial examples under other types of constraints.

- **\( L_p \) bound attacks.** To guarantee the perceptual similarity between the adversarial example and the natural example, the perturbation is normally constrained within an \( l_p \) norm bound around the natural example. To find such perturbation, Projected Gradient Descent (PGD) adversarial attack \[243\] tries to calculate the adversarial example \( x' \) that maximizes the loss function:

\[
\begin{align*}
\text{maximize} & \quad L(\theta, x') \\
\text{subject to} & \quad ||x' - x||_p \leq \epsilon \text{ and } x' \in [0, 1]^m
\end{align*}
\]

Here, \( \epsilon \) is the perturbation budget and \( m \) is the dimension of the input sample \( x \). This local maximum is calculated by the projected gradient ascent algorithm. At each time step, a small gradient step is made towards the direction of increasing loss value before the new example is projected back to the \( L_p \) norm bound. A representative attack method to achieve a standardized evaluation of robustness is called Autoattack \[104\], which is a strong evasion attack conducted by assembling four attacks, including three white-box attacks and one black-box attack. This brings a more reliable evaluation of model robustness under evasion attacks.

There are also evasion attacks with special goals. The work \[263\] devises an algorithm that successfully misleads a classifier’s decision on almost all test images. It tries to find a perturbation \( \delta \) under a \( \epsilon \) constraint satisfying that for any sample \( x \) from the test distribution, such a perturbation \( \delta \) can misguide the classifier to give wrong decisions on most of the samples.

- **Beyond \( l_p \) bound attacks.** Recently people started to realize that \( l_p \) norm perturbation budget is neither sufficient to cover real-world noise nor a perfect measurement for perceptual similarity. Some studies seek to find the minimal perturbation necessary to change the class of a given input with respect to the \( l_p \) norm \[103\]. Other works propose different perturbation measurements, e.g., the Wasserstein distance \[358, 360\], to measure the changes of pixels.

3.2.2 Poisoning Attacks. As we introduced, poisoning attacks allow adversaries to take control of the training process.

- **Training Time Attack.** In the training time attack setting, perturbation only happens during the training time. For example, the “poisoning frog” attack inserts an adversarial image with the true label to the training set, to make the trained model wrongly classify target test samples \[313\]. It generates the adversarial example \( x' \) by solving the following problem:

\[
x' = \arg\min_{x} ||Z(x) - Z(x_t)||_2^2 + \beta ||x - x_b||_2^2.
\]

Here, \( Z(x) \) is the logits of the model for samples \( x, x_t \) and \( x_b \) are the samples from the target class and the original class, respectively. The result \( x' \) would be similar to the base class in the input space while sharing similar predictions with the target class. As a concrete example, the features of birds are intentionally added into
training samples of cats but the cats are still labeled as cats in training. As a consequence, it would mislead the model’s prediction of other bird images that also contain bird features.

- **Backdoor Attack.** A backdoor attack requires that perturbations happen in both training and test data. A backdoor trigger only known by the attacker is inserted into the training data to mislead the classifier into giving a target prediction on all the test examples that contain the same trigger [84]. This type of attack is particularly dangerous because the model behaves normally on natural samples, which makes it even hard to notice the dark side of the model.

### 3.3 Representative Defense Methods

In this subsection, we introduce representative defense methods from the aforementioned categories.

#### 3.3.1 Robust Optimization / Adversarial Training.

Adversarial training aims to train models that give resistant predictions to adversarial examples. The training objective is formulated as a min-max problem that tries to minimize the error risk on the maximum adversarial loss within a small area around the training data samples [349]. With this bi-level optimization process, the model achieves partial robustness but still suffers from longer training time, natural accuracy and robust accuracy trade-offs, and robust overfitting issues. There are several studies making efforts to improve standard adversarial training from different perspectives. In [342], the trade-off issue is revealed. TRADES [386] takes a step toward balancing the natural accuracy and robust accuracy by adding a regularization term to minimize the prediction difference between adversarial samples and natural samples. Other works [314, 357] boost the training speed by estimating the gradient of the loss by treating the parameters of the last few model layers as constant layers when generating adversarial samples. They can shorten the training time to one-fourth of the GPU time – or even shorter with comparable robust performance. To mitigate robust overfitting, different classic techniques, such as early stop, weight decay, and data augmentations have been investigated [290, 359]. It is evident from recent work [71] that using data augmentation methods is a promising direction to further boost adversarial training performance.

#### 3.3.2 Certified Defense.

Certified defense seeks to learn provably robust DNNs against specific norm-bounded perturbations [286, 356]. In empirical defenses, such as adversarial training, robustness is achieved to a certain extent; in certified robust verification, however, we want to exactly answer the question of whether we can find an adversarial example for a given example. For instance, a randomized smoothing based classifier [95] aims to build an absolutely smooth classifier by making decisions according to the majority of predictions of all neighborhood examples around the original test example. Achieving such smoothness requires considerably greater computation resources, which is a challenge in practice.

#### 3.3.3 Detection.

In order to distinguish the adversarial examples in data distribution and to prevent the harmful effect, people design detection algorithms. A common way to achieve detection is to build another classifier to predict whether a sample is adversarial or not. The work [158] trains a binary classification model to discriminate all adversarial examples apart from natural samples and then builds ML models on recognized natural samples. Other works detect the adversarial samples based on the statistic property of adversarial sample distribution difference compared to natural sample distribution. Other work [162] uses a statistical test, i.e., a Maximum Mean Discrepancy (MMD) test, to determine whether two datasets are drawn from the same distribution. It uses this tool to test whether a group of data points are natural or adversarial; however, it is shown in [68] that evasion adversarial examples are not easily detected. This paper covers ten detection methods and finds out that those defenses lack thorough security evaluations and there is no clear
evidence to support that adversarial samples are intrinsically different from clean samples. Recent work [82] proposes a new direction for black-box adversarial detection. It detects the attacker’s purpose based on the historical queries and sets a threshold for the distance between two input image queries to detect suspicious attempts to generate adversarial examples.

3.4 Applications in Real Systems

When machine learning is applied to real-world, safety-critical tasks, the existence of adversarial examples becomes more dangerous and may cause severe consequences. In the following section, we illustrate the potential threats from adversarial examples to real-world applications in different domains.

3.4.1 Image Domain. In the autonomous driving domain, road sign detection is an important task. However, with some small modifications, the road sign detection system [130, 324] in the vehicle would recognize 35 mph as 85 mph and cannot successfully detect a stop sign as shown in Figure 3. Deep learning is also widely applied in authentication tasks. An attacker can wear a special glass to pretend to be an authorized identity in order to mislead the face recognition model; this deceit can be accomplished by labeling a few face samples that wear certain glasses as the target identity and by inserting those mislabeled samples into the training set [84]. Another example of identity threat is that the person detection system for images or videos can also be broken by wearing an adversarial T-shirt [364].

![Image of a stop sign that is not recognized by machines with modifications.]

Fig. 3. The stop sign could not be distinguished by machines with modifications.

3.4.2 Text Domain. Adversarial attacks also happen in natural language processing tasks, such as text classification, machine translation, and dialogue generation. For machine translation, sentence and word paraphrasing on input texts are conducted to craft adversarial examples [220]. The attacker first builds a paraphrasing corpus that contains a lot of words and sentence paraphrases. To find an optimal paraphrase of an input text, a greedy method is adopted to search for valid paraphrases for each word or sentence from the corpus. Moreover, it proposes a gradient-guided method to improve the efficiency of the greedy search. Another work [230] treats the neural dialogue model as a black box and adopts a reinforcement learning framework to effectively find trigger inputs for targeted responses. This type of black-box setting is stricter but more realistic, while the requirements for the generated responses are properly relaxed. The generated responses are expected to be semantically identical to the targeted ones but not necessarily exactly match them.
3.4.3 Audio Data. The state-of-art speech-to-text transcription networks, such as DeepSpeech [168], can be attacked by a small perturbation [70]. Given any speech waveform \( x \), an inaudible sound perturbation \( \delta \) is added to make the synthesized speech \( x + \delta \) recognized as any targeted desired phrase. In this work [303], authors propose an adversarial attack method against the YouTube CopyRight detection system to avoid music with copyright issues being detected during uploading. The attack uses a neural network to extract features from the music piece to create a fingerprint, which is used for checking whether the music matches existing copyrighted music. A gradient-based adversarial attack on the original audio piece can create a large difference in the output fingerprint such that the modified audio can successfully avoid the detection of YouTube’s CopyRight detection system.

3.4.4 Graph Data. Zügner et al. [404] considers attacking node classification models, graph convolutional networks [208], by modifying the node connections or node features. In this setting, an adversary is allowed to add or remove edges between nodes or change the node features with a limited number of operations in order to mislead the GCN model that is trained on the perturbed graph. The work [405] attempts to poison the graph so that the global node classification performance of GCN will drop – and even be made almost useless. They optimize the graph structure as the hyper-parameters of the GCN model with the meta-learning technique. The goal of this attack [44] is to perturb the graph structure to corrupt the quality of node embedding, affecting the downstream tasks’ performance, including node classification or link prediction. Pro-GNN [191] tries to jointly learn a structural graph and a graph neural network from the perturbed graph guided by some intrinsic properties of a real-world graph, such as low-rank and feature smoothness. The defense method could learn a clean adjacency matrix close to the perturbed one and limit its norm to guarantee the low-rank property. Meanwhile, a feature smoothness regularizer is also utilized to penalize rapid changes in features between adjacent nodes. Then a robust GNN would be built upon the learned graph.

3.5 Robustifying Real World System

To make real-world ML systems safer, many strategies can be applied. As we mentioned, the Youtube music copyright system can be fooled by simply adding some unnoticeable noise. With the help of adversarial training, the problem could be eased [303]. Also, a chatbot is a commonly used machine learning system, and inappropriate language can be a safety problem. To mitigate the harmful effect, the work [366] proposes a robust evaluation and enhancement method to improve the reliability of dialogue systems. Meanwhile, people are seeking methods to create adversarial autonomous driving environments to enhance the safety of autonomous driving.

3.6 Surveys and Tools

In this subsection, we list related resources about adversarial robustness, including surveys and tools.

3.6.1 Surveys. Xu et al. [364] gives a comprehensive introduction of concepts and covers representative attack and defense algorithms in different domains, including image classification, graph classification, and natural language processing. For the surveys in a specific domain, Akhtar and Mian [25] provides a comprehensive introduction to adversarial threats in a computer vision domain [76]; Jin et al. [190] gives a thorough review of the latest adversarial robustness techniques in the graph domain; and Zhang et al. [389] focuses on natural language processing and summarizes important algorithms on adversarial robustness in the text domain.

3.6.2 Tools. Advertorch [118] is a Pytorch toolbox containing popular attack methods in the image domain. DeepRobust [226] is a comprehensive and up-to-date adversarial attacks and defenses library based on Pytorch that includes not
only algorithms in the image domain but also the graph domain. This platform provides convenient access to different algorithms and evaluation functions to illustrate the robustness of image classification models or graph properties. RobustBench [102] provides a robust evaluation platform by the Autoattack algorithm for different adversarial training models. This platform also provides well-trained robust models by different adversarial training methods, which can save resources for researchers. A summary of the above-mentioned open-source toolkits and frameworks on robustness in AI can be found in Table 2.

### Table 2. Representative open-source toolkits and frameworks on robustness in AI.

<table>
<thead>
<tr>
<th>Toolkit</th>
<th>Characteristics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cleverhans [275]</td>
<td>A Tensorflow library for benchmarking the vulnerability of machine learning models to adversarial examples.</td>
</tr>
<tr>
<td>DeepRobust [226]</td>
<td>A Pytorch toolbox containing attack and defense methods in image and graph domain.</td>
</tr>
<tr>
<td>RobustBench [102]</td>
<td>A tool to standardize the evaluation of adversarial robustness.</td>
</tr>
</tbody>
</table>

#### 3.7 Future Directions

For adversarial attacks, people are seeking more general attacking methods to evaluate adversarial robustness. For black-box attacks, efficiently generating adversarial examples with fewer adversarial queries is often challenging. For training in adversarial defense methods, an important issue is the robust overfitting and lack of generalization in both adversarial and natural examples. These problems remain unsolved and need further improvements. Another direction for adversarial training is to build robust models against more general adversarial examples, including but not limited to different l-p bound attacks. For certified defenses, one possible direction is to train a model with robust guarantees more efficiently, since the current certified defense methods require a large number of computational resources.

#### 4 NON-DISCRIMINATION & FAIRNESS

As AI plays an increasingly irreplaceable role in various scenarios closely related to people’s vital interests, such as recidivism prediction, financial risk assessment and job recommendation, an AI system ought to avoid discriminatory behaviors in human-machine interactions and ensure fairness in decision making for any individuals or groups. Otherwise, it would lose the trust from various stakeholders. With the rapid spread of AI in our daily lives, more and more evidence indicates that AI systems show human-like discriminatory bias or make unfair decisions. For example, a recidivism prediction software used by U.S. courts often assigns a higher risky score for an African American than a Caucasian with a similar profile [253]; a job recommendation system promotes more STEM employment opportunities to male candidates than to females [215]. Moreover, Tay, the online AI chatbot developed by Microsoft, produced a lot of improper racist and sexist comments, which led to its closure within 24 hours of release [354]; dialogue models trained on human conversations show bias toward females and African Americans by generating more offensive and negative responses for these groups [229]. Fairness in AI demands considerable attention. Recently, many works have emerged to define, recognize, measure, and mitigate the bias in AI algorithms. In this section, we aim to give a comprehensive overview of the cutting-edge research progress addressing fairness issues in AI. In the subsections, we first present concepts and definitions regarding fairness in AI. We then provide a detailed taxonomy to discuss

different origins of algorithmic bias, different types of bias, and fairness. We then review and classify popular bias mitigation technologies for building fair AI systems. Next, we introduce the specific bias issues and the applications of bias mitigation methods in real-world AI systems. In this part, we categorize the works according to the types of data processed by the system. Finally, we discuss the current challenges and future opportunities in this field. We expect that researchers and practitioners can gain a sense of direction and understanding from a broad overview of bias and fairness issues in AI and deep insight into the existing solutions, so as to advance progress in this field.

4.1 Concepts and Taxonomy

Before we go deep into nondiscrimination and fairness in AI, we need to understand how relative concepts, such as bias and fairness, are defined in this context. In this subsection, we briefly illustrate the concepts of bias and fairness, and provide a taxonomy to introduce different sources of bias, different types of bias, and fairness.

4.1.1 Bias. In the machine learning field, the word “bias” has been abused. It conveys different meanings in different contexts. We first distinguish the concept of bias in the context of AI non-discrimination and fairness from that in other contexts. There are three categories of bias: productive bias, erroneous bias, and discriminatory bias. Productive bias exists in all machine learning algorithms. It is beneficial and necessary for an algorithm to be able to model the data and make decisions [175]. Based on the “no free lunch theory” [355], only a predictive model biased toward certain distributions or functions can achieve better performance on modeling them. Productive bias helps an algorithm to solve certain types of problems. It is introduced through our assumptions about the problem, which is specifically reflected as the choice of a loss function, an assumed distribution, or an optimization method, etc. Erroneous bias can be viewed as a systematic error caused by faulty assumptions. For example, we typically assume that the distribution of the training data is consistent with the real data distribution. However, due to selection bias [244] or sampling bias [253], the collected training data may not be able to represent the real data distribution. Thus, the violation of our assumption can lead to the learned model’s undesirable performance on the test data. Discriminatory bias is the kind of bias we are interested in under AI nondiscrimination and fairness. As opposed to fairness, discriminatory bias reflects an algorithm’s unfair behaviors toward a certain group or individual, such as producing discriminatory content or performing less well for some people [316]. In the rest of this paper, when we mention bias, we refer to discriminatory bias.

Sources of Bias. The bias in an AI system can be produced by different sources, namely, the data, the algorithm, or the evaluation method. Bias within data comes from different phases of data generation, from data annotation to data collection and data processing [272, 315]. In the phase of data annotation, bias can be introduced due to a non-representative group of annotators [192], inexperienced annotators [278], or preconceived stereotypes held by the annotators [309]. In the phase of data collection, bias can emerge due to the selection of data sources or how data from several different sources are acquired and prepared [272]. In the data processing stage, bias can be generated due to data cleaning [115], data enrichment [96], and data aggregation [343].

Types of Bias. Bias can be categorized into different classes from different perspectives. It can be explicit or implicit. Explicit bias, also known as direct bias, occurs when the sensitive attribute explicitly causes an undesirable outcome for an individual; while implicit bias, also known as indirect bias, indicates the phenomenon that an undesirable outcome is caused by nonsensitive and seemingly neutral attributes, which in fact have some potential associations with the sensitive attributes [387]. For example, the residential address seems a nonsensitive attribute, but it can correlate with the race of a person according to the population distribution of different ethnic groups [387]. Moreover, language
style can reflect the demographic features of a person, such as race and age [182, 231]. Bias can be acceptable and unacceptable. Acceptable bias, also known as explainable bias, describes a situation where the discrepancy of outcomes for different individuals or groups can be reasonably explained by factors. For example, models trained on the UCI Adult dataset predict higher salaries for males than females. Actually, this is because males work for a longer time per week than females [200]. Based on this fact, such biased outcomes are acceptable and reasonable. Conversely, bias that cannot be explained appropriately is treated as unacceptable bias, which should be avoided in practice.

4.1.2 Fairness. The fairness of an algorithm is defined as "the absence of any prejudice or favoritism toward an individual or a group based on their intrinsic or acquired traits in the context of decision making" [253, 310]. Furthermore, according to the object of the study, fairness can be further defined as group fairness and individual fairness.

Group Fairness. Group fairness requires that two groups of people with different sensitive attributes receive comparable treatments and outcomes statistically. Based on this principle, various definitions have been proposed, such as Equal Opportunity [169], which requires people from two groups to be equally likely to get a positive outcome when they indeed belong to the positive class; Equal Odds [169], which requires that the probability of being classified correctly should be the same for different groups; and Demographic Parity [123], which requires different groups to have the same chance to get a positive outcome, etc.

Individual Fairness. While group fairness can maintain fair outcomes for a group of people, a model can still behave discriminatorily at the individual level [123]. Individual fairness is based on the understanding that similar individuals should be treated similarly. A model satisfies individual fairness if it gives similar predictions to similar individuals [123, 213]. Formally, if individuals $i$ and $j$ are similar under a certain metric $\delta$, the difference between the predictions given by an algorithm $M$ on them should be small enough: $|f_M(i) - f_M(j)| < \epsilon$, where $f_M(\cdot)$ is the predictive function of algorithm $M$ that maps an individual to an outcome, and $\epsilon$ is a small constant.

4.2 Methods

In this subsection, we introduce bias mitigation techniques. Based on which stage of an AI pipeline is to interfere, the debiasing methods can be categorized into three types: pre-processing, in-processing and post-processing methods. Representative bias mitigation methods are summarized in Table 3.

<table>
<thead>
<tr>
<th>Category</th>
<th>Strategy</th>
<th>References</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pre-processing</td>
<td>Sampling</td>
<td>[20, 35, 394]</td>
</tr>
<tr>
<td></td>
<td>Reweighting</td>
<td>[64, 199, 385]</td>
</tr>
<tr>
<td></td>
<td>Blinding</td>
<td>[79, 94, 170, 380]</td>
</tr>
<tr>
<td></td>
<td>Relabelling</td>
<td>[101, 164, 199]</td>
</tr>
<tr>
<td></td>
<td>Adversarial Learning</td>
<td>[19, 141, 195]</td>
</tr>
<tr>
<td>In-processing</td>
<td>Reweighting</td>
<td>[188, 211]</td>
</tr>
<tr>
<td></td>
<td>Regularization</td>
<td>[22, 139]</td>
</tr>
<tr>
<td></td>
<td>Bandits</td>
<td>[126, 236]</td>
</tr>
<tr>
<td></td>
<td>Adversarial Learning</td>
<td>[75, 231, 232, 384]</td>
</tr>
<tr>
<td>Post-processing</td>
<td>Thresholding</td>
<td>[169, 184, 254]</td>
</tr>
<tr>
<td></td>
<td>Transformation</td>
<td>[89, 205, 266]</td>
</tr>
<tr>
<td></td>
<td>Calibration</td>
<td>[172, 286]</td>
</tr>
</tbody>
</table>
Pre-processing Methods. Pre-processing approaches try to remove the bias in the training data to ensure the fairness of an algorithm from the origin [199]. This category of methods can be adopted only when we have access to the training data. Various strategies are proposed to interfere with training data. Specifically, Celis et al. [74] propose to adaptively sample the instances that are both diverse in features and fair to sensitive training attributes. Moreover, reweighting methods [198, 385] try to mitigate the bias in training data by adaptively up-weighting the training instances of underrepresented groups, while down-weighting those of overrepresented groups. Blinding methods try to make a classifier insensitive to a protected variable. For example, Hardt et al. [170] force a classifier to have the same threshold value for different race groups to ensure that the predicted loan rate is equal for all races. Some works [198, 382] try to relabel the training data to ensure the proportion of positive instances are equal across all protected groups. Additionally, Xu et al. [362] take advantage of a generative adversarial network to produce bias-free and high-utility training data.

In-processing Methods. In-processing approaches address the bias at the algorithm level and try to eliminate bias during the model training process. They often seek to create a balance between performance and fairness [73]. Krasanakis et al. [211] propose an in-processing re-weighting approach. They first train a vanilla classifier to learn the weights of samples and then retrain the classifier using these weights. Some works [154, 201] take advantage of regularization methods, where one or more penalty terms are added into the objective function to penalize biased outcomes. The idea of adversarial learning is also adopted in in-processing debiasing methods. Liu et al. [232] design an adversarial learning framework to train neural dialogue models that are free from gender bias. Alternatively, bandits recently have emerged as a novel idea for solving fairness problems. For example, Joseph et al. [193] propose solving the fairness problem under a stochastic multi-armed bandit framework with fairness metrics as the rewards and the individuals or groups under investigation as the arms.

Post-processing Methods. Post-processing approaches directly make transformations on the model’s outputs to ensure fair final outcomes. Hardt et al. [170] propose approaches to determine threshold values via measures such as equalized odds specifically for different protected groups to find a balance between the true and false positive rates to minimize the expected classifier loss. Feldman et al. [138] propose a transformation method to learn a new fair representation of the data. Specifically, they transform the SAT score into a distribution of the rank order of the students independent of gender. Pleiss et al. [279] borrow the idea of calibration to build fair classifiers. Similar to the traditional definition of calibration that the proportion of positive predictions should be equal to the proportion of positive examples, they force the conditions to hold for different groups of people. Nevertheless, they also find that there is a tension between prediction accuracy and calibration.

4.3 Applications in Real Systems

In this subsection, we summarize the studies regarding bias and fairness issues in real-world AI systems during different tasks. We introduce the works following the order of different data domains, including tabular data, images, texts, audios, and graphs. For each domain, we describe several representative tasks and present how AI systems can be biased on these tasks. A summary of the representative works can be found in Table 4. In addition, we demonstrate various examples of adopting the debiasing methods from Section 4.2 to mitigate the bias in real-world AI systems.

4.3.1 Tabular Domain. Tabular data is the most common format of data in machine learning; thus, the research on bias in machine learning is predominantly conducted on tabular data. In the recent decade, researchers have investigated how algorithms can be biased in classification, regression, and clustering tasks. For classification, researchers find evidence
that machine learning models for credit prediction [197] and recidivism prediction [93] tasks can show significant prejudice toward certain demographic attributes of a person, such as race and gender. Berk et al. [38] and Agarwal et al. [21] investigate multiple regression tasks, from salary estimation to crime rate prediction, showing unfair treatment for different races and genders. Backurs et al. [31] and Chen et al. [83] discover unfairness in clustering algorithms with a belief that as data points, different groups of people are entitled to be clustered with the same accuracy.

4.3.2 Image Domain. Machine learning models in computer vision have also shown unfair behaviors. In [59, 178], the authors show that face recognition systems work better for white compared to darker faces, and also show similar bias in terms of gender. An image classification application developed by Google has been accused of labeling black people as "gorillas" [273]. In [302], the authors discover the difference in the performances of smiling detection models on different genders and races. The work [397] tackle the social bias in visual semantic role labeling, e.g., associating cooking roles with women. They introduce corpus-level constraints for calibrating existing structured prediction models. In the work [349], a visual recognition benchmark is designed for studying bias mitigation.

4.3.3 Text Domain. A large number of works have shown that algorithmic bias exists in various natural language processing tasks. Word embeddings often exhibit a stereotypical bias for text data, causing a serious risk of perpetuating problematic biases in imperative societal contexts. In [47], the authors first shows that popular state-of-the-art word embeddings regularly mapped men to working roles and women to traditional gender roles, leading to significant gender bias which is even inherited in downstream tasks. Following the research of word embeddings, the same patterns of gender bias are discovered in sentence embeddings [247]. In the task of co-reference resolution, researchers demonstrate in [398] that rule-based, feature-based, and neural network-based co-reference systems all show gender bias by linking gendered pronouns to pro-stereotypical entities with higher accuracy than anti-stereotypical entities. Language models can also learn gender discrimination from man-made text data [49], which tend to generate certain words reflecting gender stereotypes with different probabilities in the context of males and females. As for machine translation, it has been illustrated that Google’s translation system suffers from gender bias by showing favoritism toward males for stereotypical fields, such as STEM jobs when translating sentences taken from the U.S. Bureau of Labor Statistics into a
dozen gender-neutral languages [281]. Dialogue systems, including generative models and retrieval-based models, also show bias toward different genders and races by producing discriminatory responses [229, 232].

4.3.4 Audio Domain. Voice recognition systems show gender bias by processing the voices of men and women differently [178]. It is found that medical voice-dictation systems recognize voice inputs from males versus females with higher accuracy [294]. It is shown in [72] that voice control systems on vehicles worked better for males than females. Google’s speech recognition software can understand queries from male voices more consistently than those from females [336].

4.3.5 Graph Domain. AI applications on graph-structured data are ubiquitous in the real world. The fairness issues in these problems are drawing increasing attention from researchers. Existing graph embedding techniques can learn node representations correlated with protected attributes, such as age and gender. Consequently, they exhibit bias toward certain groups in real-world applications, like social network analysis and recommendations [51]. Graph neural networks (GNNs) also inherit bias from training data and even magnify the bias through GNN’s graph structures and message-passing mechanism [107].

4.3.6 Debiasing Real Systems. Many attempts have been made to ensure fairness in real-world AI systems. Zhang et al. [384] utilizes the in-processing adversarial debiasing strategy to train a fair classification model. During the training process, an adversary is introduced to judge whether the outcomes are fair or not, and the feedback from the adversary will serve as the guidance to improve the fairness of the classifier. The work [178] introduces a pre-processing method where training data is balanced to alleviate gender bias in face recognition tasks. In terms of text data, Zhang et al. [385] develops an instance weighting framework that can effectively mitigate the bias of text classifiers toward demographic identity-terms, while the prediction capability of the models is not significantly affected.

4.4 Surveys and Tools
In this subsection, we gather the existing surveys, tools and repositories on fairness in AI to facilitate readers wishing to explore this field further.

4.4.1 Surveys. The problem of fairness has been studied in multiple disciplines other than computer science for more than a half century. In one survey [183], the authors trace the evolution of the notions and measurements of fairness in different fields, such as education and hiring, over the past 50 years. They provide a comprehensive comparison between the past and current definitions to encourage a deeper understanding of modern fairness in AI. Zliobaite [402] provides an early survey on measuring indirect discrimination in machine learning. In this survey, the authors review early approaches for measuring bias in data and predictive models. They also analyze the measurements from other fields and explore the possibility of their use in machine learning. Corbett-Davies and Goel [99] provide a critical review on the measurements of fairness, showing the limitations of the existing fairness criteria in classification tasks in machine learning. Mehrabi et al. [253] contribute a comprehensive survey on bias and fairness in machine learning. In this survey, the authors provide a detailed taxonomy of the bias and fairness definitions in machine learning, and also introduce the bias observed in the data and algorithms in different domains of AI and the state-of-the-art debiasing methods. Caton and Haas [73] provide an overview of the existing debiasing approaches for building fair machine learning models. They organize the extant works into three categories and 11 method areas and introduce them following their taxonomy. Moreover, there are some surveys regarding bias and fairness in specific domains of AI. Blodgett et al. [43] review the papers analyzing bias in NLP systems, providing critical comments on such works and indicating that many
existing works suffer from unclear and inconsistent motivations and irrational reasoning. They also offer suggestions to normalize future studies on bias in NLP. Chen et al. [80] summarize and organize the works on bias and debias in recommender systems, and discuss future directions in this field.

4.4.2 Tools. In recent years, some organizations or individual researchers have provided multi-featured toolkits and repositories to facilitate fair AI. The repository Responsibly [237] collects the datasets and measurements for evaluating bias and fairness in classification and NLP tasks. The project FairTest [339] provides an unwarranted associations (UA) framework to discover unfair user treatment in data-driven algorithms. AIF360 [36] collects popular datasets for fairness studies and provides the implementations of common debiasing methods for binary classification. Aequitas [307] is released as an audit toolkit to test the bias and fairness of models for multiple demographic groups on different metrics. The repository Fairness Measurements provides datasets and codes for quantitatively measuring discrimination in classification and ranking tasks. A summary of the above-mentioned open-source toolkits and frameworks on fairness in AI can be found in Table 5.

<table>
<thead>
<tr>
<th>Toolkit</th>
<th>Characteristics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Responsibly [237]</td>
<td>Datasets and measurements for bias and fairness evaluation on classification tasks and NLP tasks.</td>
</tr>
<tr>
<td>AIF360 [36]</td>
<td>Popular datasets for fairness research and implementations of common debiasing algorithms.</td>
</tr>
<tr>
<td>Aequitas [307]</td>
<td>An audit framework for testing the fairness of ML models.</td>
</tr>
<tr>
<td>Fairness Measurements</td>
<td>Datasets and implementations for measuring bias in classification and ranking tasks.</td>
</tr>
</tbody>
</table>

4.5 Future Directions

Fairness research still possesses a number of outstanding challenges.

- **Trade-off between fairness and performance.** Studies on fairness in different fields have confirmed the existence of the trade-off between fairness and performance of an algorithm [39, 100, 283]. The improvement of the fairness of an algorithm typically comes at the cost of performance degradation. Since both fairness and performance are indispensable, extensive research is needed to help people better understand an algorithm’s trade-off mechanism between them, so that practitioners can adjust the balance in practical usage based on the actual demand;

- **Precise conceptualization of fairness.** Although extensive research has been conducted on bias and fairness in AI, too much of this work formulates its concerns under a vague concept of bias that refers to any system harmful to human behaviors but fails to provide a precise definition of bias or fairness specific to their setting [43]. In fact, different forms of bias can appear in different tasks, even in the same task. For example, in a recommender system, popularity bias can exist toward both the users and items [80]. In a toxicity detection algorithm, race bias can exist toward both the people mentioned in texts and the authors of texts [231]. To study any fairness problem, a precise definition of bias indicating how, to whom, and why an algorithm can be harmful must be articulated. In this way, we can make the research on AI fairness in the whole community more standardized and systematic;
• **From equality to equity.** Fairness definitions are often associated with equality to ensure that an individual or a conserved group, based on race or gender, are given similar amounts of resources, consideration, and results. Nonetheless, the area of equity has been heavily under-examined [253], where this notion pertains to the particular resources for an individual or a conserved group to be successful [159]. Equity remains an interesting future direction, since the exploration of this definition can extend or contradict existing definitions of fairness in machine learning.

5 EXPLAINABILITY

Making the AI techniques transparent can enhance the trustworthiness, where humans can understand how/why it can work well and when/why/where it doesn’t work, so that users can adopt the AI techniques accordingly and fully trust them afterward. The improved predictive performance of AI systems has often been achieved through increased model complexity [120, 262]. A prime example is the paradigm of deep learning, dominating the heart of most state-of-the-art AI systems. However, deep learning models are treated as black-boxes, since most of them are too complicated and opaque to be understood and are developed without explainability [227]. More importantly, without explaining the underlying mechanisms behind the predictions, deep models cannot be fully trusted, which prevents their use in critical applications pertaining to ethics, justice, and safety, such as healthcare [258], autonomous cars [221], and so on. Therefore, building a trustworthy AI system requires an understanding of how particular decisions are made [144], which has led to the revival of the field of eXplainable Artificial Intelligence (XAI). In this section, we aim to provide an intuitive understanding and high-level insights into the recent progress of explainable AI. First, we provide the concepts and taxonomy regarding explainability in AI. Second, we review representative explainable techniques for AI systems according to the aforementioned taxonomy. Third, we introduce real-world applications of explainable AI techniques. Finally, we provide some surveys and tools and discuss future opportunities for explainable AI.

5.1 Concepts and Taxonomy

In this subsection, we introduce the concepts of explainability in AI. We then provide a taxonomy of different explanation techniques.

5.1.1 Concepts. In the context of machine learning and AI literature, explainability and interpretability are usually used by researchers interchangeably [262]. One of the most popular definitions of explainability is the one from Doshi-Velez and Kim, who define it as “the ability to explain or to present in understandable terms to a human” [120]. Another popular definition is from Miller, who defines explainability as “the degree to which a human can understand the cause of a decision” [257]. In general, the higher the explainability of an AI system is, the easier it is for someone to comprehend how certain decisions or predictions have been made. Meanwhile, a model is better explainable than other models if its decisions are easier for a human to comprehend than those of others.

While explainable AI and interpretable AI are very closely related, subtle differences between them are discussed in some studies [153, 300, 391].

• A model is interpretable if the model itself is capable of being understood by humans on its predictions. When looking at the model parameters or a model summary, humans can understand exactly the procedure on how it made a certain prediction/decision and, even given a change in input data or algorithmic parameters, it is the extent to which humans can predict what is going to happen. In other words, such models are intrinsically
transparent and interpretable, rather than black-box/opaque models. Examples of interpretable models include decision trees and linear regression.

- An explainable model indicates that additional (post hoc) explanation techniques are adopted to help humans understand why it made a certain prediction/decision that it did, although the model is still black-box and opaque. Note that such explanations are often not reliable and can be misleading. Examples of such models would be deep neural network based models, where the models are usually too complicated for any human to comprehend.

5.1.2 Taxonomy. Techniques for AI’s explanation can be grouped according to various criteria such as model, scope and method.

- **By Model: Intrinsic or Post-hoc.** If AI models are considered transparent or interpretable due to their simple structures, then these interpretable techniques are called a model intrinsic explanation. In contrast, post-hoc methods refer to the explainability methods that are developed to explain the target AI model after model training.

- **By Scope: Local or Global.** If the method provides an explanation only for a specific instance, then it is a local explanation; if the method explains the whole model, then it is a global explanation.

- **By Method: Gradient-based, Perturbation-based, or Others.** Most existing explainability methods can be categorized into two main classes: gradient-based and perturbation-based methods. If the techniques employ the partial derivatives on input instances to generate attributions, then these techniques are called a gradient-based explanation. If the techniques focus on the changes or modifications of input data, we name them a perturbation-based explanation. In addition, there are other explanation techniques beyond gradient/perturbation-based methods. For example, a counterfactual explanation usually refers to a causal situation in the form, “If \( X \) had not occurred, \( Y \) would not have occurred.” In general, counterfactual explanation methods are post-hoc and can be used to explain predictions of individual instances (local) [262, 346].

5.2 Methods

In this subsection, we introduce some representative explanation techniques according to the aforementioned taxonomy. A summary of the representative works can be found in Table 6.

5.2.1 **By Model: Model-intrinsic or Post-hoc.** Any explainable algorithm that is dependent on the model architecture can fall into the model-intrinsic category. In contrast, post-hoc methods apply to any model for being generally applicable. In general, there are significant research interests in developing post-hoc methods to explain the predictions of an existing well-performing neural networks model. This criterion also can be used to distinguish whether interpretability is achieved by restricting the complexity of the AI model. Intrinsic interpretability refers to AI models that are considered interpretable (white-box) due to their simple model architecture, while most post-hoc explanations are widely applied into (black-box) deep neural networks which are highly complicated and opaque due to their millions of parameters.

- **Model-intrinsic Explanations.** The model in this category is often called an intrinsic, transparent, or white-box explanation. Generally, without designing an additional explanation algorithm, this type of interpretable technique cannot be re-used by other classifier architectures. Therefore, the model intrinsic methods of explanations are inherently model specific. Such commonly used interpretable models include linear/logistic regression, decision trees, rule-based models, Generalized Additive Models (GAMs), Bayesian networks, etc.
For example, the linear regression model [42], which is one of the most representative linear models in ML, aims to predict the target as a weighted sum of the feature of instances. With this linearity of the learned relationship, the linear regression model makes the estimation procedure simple and significantly understandable on a modular level (i.e., the weights) for humans. Mathematically, given one instance with $d$ dimension of features $x$, the linear regression model can be used to model the dependence of a predicted target $\hat{y}$ as follows:

$$\hat{y} = w^T x + b = w_1 x_1 + ... + w_d x_d + b$$

(1)

where $w$ and $b$ denote the learned feature weights and the bias term, respectively. The predicted target $\hat{y}$ of linear regression is a weighted sum of its $d$ dimension features $x$ for any instance, where the decision-making procedure is easy for a human to comprehend by inspecting the value of the learned feature weights $w$.

Another representative method is decision tree [284], which contains a set of conditional statements arranged hierarchically. Making predictions in a decision tree is also the procedure of explaining the model by seeking the path from the root node to leaf nodes (label), as illustrated in Figure 4.

Fig. 4. The interpretation of decision tree is simple, where intermediate nodes in the tree represent decisions and leaf nodes can be class labels. Starting from the root node to leaf nodes can create good explanations on how the certain label is made by decision tree model. (Image Credit: [262])

Fig. 5. LIME explains deep neural networks on image classification task: top 3 predicted categories and the corresponding scores. (Image Credit: [289])
- **Post-hoc Explanations.** The methods in this category are concerned with black-box well-trained AI models. More specifically, these methods do not try to create interpretable models, but to interpret already well-trained models. Such methods have been widely used for explaining complicated models, such as deep neural networks. That is why they sometimes are referred to as black-box explainability methods in the related scientific literature. One advantage of post-hoc methods over model-intrinsic ones is their flexibility. Post-hoc methods have also been applied in a variety of input modalities, such as images, text, graph-structured data, etc. Note that post-hoc methods can also be applied to intrinsically interpretable models.

One of the most representative works in this category is Local Interpretable Model-Agnostic Explanations (LIME) [289]. For example, at the image domain, for any trained classifier, LIME is a proxy approach to randomly permute data by identifying the importance of local contiguous patches with similar pixels in a given instance and its corresponding label [289]. An illustrative example of LIME on a single instance for the top three predicted classes is shown in Figure 5.

Additionally, to understand how any graph neural networks (GNNs) make a certain decision on graph-structured data, GNNExplainer learns soft masks for edges and node features to explain the predictions via maximizing the mutual information between the predictions of the original graph and those of the newly obtained graph [240, 372]. Figure 6 illustrates explanation examples generated by GNNExplainer for graph-structured data.
5.2.2 **By Scope: Local or Global.** One important aspect of dividing the explainability techniques is based on the scope of explanation, i.e., local or global.

- **Local Explanations.** In general, the goal of locally explainable methods is to express the individual feature attributions of a single instance of input data $x$ from the data population $X$. For example, given a text document and a model to understand the sentiment of text, a locally explainable model might generate attribution scores for individual words in the text.

  In the Saliency Map Visualization method [323], the authors compute the gradient of the output class category with regard to an input image. By visualizing the gradients, a fair summary of pixel importance can be achieved by studying the positive gradients that have more influence on the output [323]. An example of the class model is shown in Figure 7.

- **Global Explanations.** The goal of global explanations is to provide insights into the decision of the model as a whole and to have an understanding of attributions for a batch of input data or a certain label, not just for individual inputs. In general, globally explainable methods work on an array of inputs to summarize the overall behavior of the black-box model. Most linear, rule-based and tree-based models are inherently globally explainable. For example, conditional statements (intermediate nodes) in decision trees can give insight into how the model behaves in a global view, as shown in Figure 4.

  In terms of the DNNs models, Class Saliency Map Visualization [323] is trying to generate a particular image visualization by maximizing the score of class probability with respect to the input image (i.e., using the derivative of predicted class score with respect to the image). An example of the class model is shown in Figure 8. SP-LIME extends the vanilla LIME method (which provides local explanations on an individual instance) to give a global understanding via submodular optimization [289].

5.2.3 **By Method: Gradient-based, Perturbation-based, or Others.** This category is mainly defined by answering the question, "What is the algorithmic approach? Does it focus on the input data instance or the model parameters?" Based on the core algorithmic approach of the explanation method, we can categorize explanation methods as the ones that focus on the gradients of the target prediction with respect to input data, and those that focus on the changes or modifications of input data. Afterward, we also introduce other representative explanation techniques beyond gradient-based and perturbation-based methods.

- **Gradient-based Explanations.** In gradient-based methods, the explainable algorithm does one or more forward passes through the neural networks and generates attributions during the back-propagation stage utilizing partial
Grad-CAM visualizations are highly interpretable and easily the discriminative image regions in a single forward-ability until the final layer. This tweaking allows identifying beyond simply acting as a regularizer - In fact, with a little fitting during training. In our experiments, we found that the Network in Network (NIN)\cite{16}: fully-connected layers are used for classification. Recently, networks (CNNs) actually behave as object detectors de-volutional units of various layers of convolutional neural networks (CNN)\cite{25}.

The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the U.S. Government, or any other organization.

**Abstract**
Learning Deep Features for Discriminative Localization
Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, Antonio Torralba

**1. Introduction**
To understand how a CNN model makes decisions, Class activation mapping (CAM)\cite{2} proposes modifying fully connected layers of the original CNN architecture using Global Average Pooling and generating important class-specific regions of the image for visual explanations via the forward passes process. An illustrative example is provided in Figure 9. Afterward, Gradient-weighted Class Activation Mapping (Grad-CAM)\cite{31} generalizes the CAM model for any CNN model without requiring architectural changes or retraining and utilizes the gradient signal flowing into the final convolutional layer of a CNN for highlighting the important regions in the image. Figure 10 shows a visual explanation via Grad-CAM for an image captioning model.

**Perturbation-based Explanations.** Perturbation-based explainable methods focus on variations in the input feature space to explain individual feature attributions toward the output class. More specifically, explanations are generated by iteratively probing a trained AI model with different variations of the inputs. These perturbations can be on a feature level by replacing certain features with zero or random counterfactual instances, picking one or a group of pixels (super-pixels) for the explanation, blurring, shifting, or masking operations, etc. In general, only a forward pass is sufficient to generate the attribution representations without the need for back-propagating.
gradients. Shapley Additive explanations (SHAP) [239] visualizes feature interactions and feature importance by probing feature correlations by removing features in a game-theoretic framework.

**Others.** Here, we present other explanation techniques that cannot be easily categorized into gradient/perturbation-based methods. For example, counterfactual explanations have been designed to answer hypothetical questions and describe how altering feature values of an instance would change the prediction to a predefined output [260, 262]. Taking the application for a credit card as an example, Peter gets rejected by AI banking systems and wonders why his application was rejected. To answer the question of “why,” counterfactual explanations can be formulated as “What would have happened to this decision (from rejected to approved), if performing minimal changes in feature values (e.g., income, age, race, etc.)?” In fact, counterfactual explanations are usually human-friendly, since they are contrastive to the current instances and usually focus on a small number of features. To generate counterfactual explanations, Wachter et. al [346] propose Lagrangian style constrained optimization as follows:

\[
\arg \min_{x} \max_{\lambda} \lambda \cdot (f(\hat{x}) - \hat{y})^2 + d(\hat{x}, x)
\]

where \(x\) is the original instance feature, and \(\hat{x}\) is the corresponding counterfactual input. \(f(\hat{x})\) is the predicted result of a classifier. The first term is the quadratic distance between the model prediction for the counterfactual \(\hat{x}\) and the targeted output \(\hat{y}\). The second term indicates the distance \(d(\cdot , \cdot )\) between the instance \(x\) to be explained and the counterfactual \(\hat{x}\), and \(\lambda\) is proposed to achieve the trade-off between the distance in prediction and the distance in feature values. Note that counterfactual explanations can be computed by gradients or perturbation methods.

Table 6. Summary of Published Research in Explainability of AI Systems.

<table>
<thead>
<tr>
<th>Representative Models</th>
<th>Intrinsic vs. Post-hoc</th>
<th>Scope</th>
<th>Methodology</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linear model</td>
<td>Intrinsic</td>
<td>Global</td>
<td>Others</td>
</tr>
<tr>
<td>LIME [289]</td>
<td>Post-hoc</td>
<td>Local</td>
<td>Perturbation</td>
</tr>
<tr>
<td>SP-LIME [289]</td>
<td>Post-hoc</td>
<td>Global</td>
<td>Perturbation</td>
</tr>
<tr>
<td>CAM [399]</td>
<td>Post-hoc</td>
<td>Local</td>
<td>Others</td>
</tr>
<tr>
<td>Grad-CAM [312]</td>
<td>Post-hoc</td>
<td>Local</td>
<td>Gradient</td>
</tr>
<tr>
<td>SHAP [239]</td>
<td>Post-hoc</td>
<td>Both</td>
<td>Perturbation</td>
</tr>
<tr>
<td>Saliency Map Visualization [323]</td>
<td>Post-hoc</td>
<td>Local</td>
<td>Gradient</td>
</tr>
<tr>
<td>Class Saliency Map Visualization [323]</td>
<td>Post-hoc</td>
<td>Global</td>
<td>Gradient</td>
</tr>
<tr>
<td>GNNExplainer [372]</td>
<td>Post-hoc</td>
<td>Local</td>
<td>Perturbation</td>
</tr>
<tr>
<td>Surveys</td>
<td>[16, 29, 37, 110, 120, 121, 163, 189, 227, 257, 262, 338, 379, 391]</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### 5.3 Applications in Real Systems

When a prediction is complemented with explanations to understand how particular decisions are made, AI systems will gain more trust from humans. Various explanations techniques are proposed to integrate into real-world AI systems for enhancing the trustworthiness. In this subsection, we discuss representative real-world applications where explainability is crucial.

#### 5.3.1 E-commerce

The explosive popularity of e-commerce sites has attracted an increasing number of people to spend more time shopping online. Recommender systems (RecSys) as intelligent systems have become increasingly
important in mitigating the information overload problem in e-commerce [134, 135]. These systems provide personalized information to help human decisions and have been widely used in various user-oriented online services [132], such as e-commerce item recommendations for everyday shopping (e.g., Amazon, Taobao), job recommendations for employment markets (e.g., LinkedIn), and friend recommendations to make people better connected (e.g., Facebook, Weibo) [131, 136]. Recently, in order to increase the trustworthiness, increasing attention has been paid to understanding why certain items have been recommended by deep learning-based recommender systems for end-users, because providing good explanations of personalized recommender systems can sufficiently motivate users to interact with items, help users make better and/or faster decisions, and increase users’ trust in the intelligent recommender systems [241, 393]. For example, to achieve explainability in recommender systems, RuleRec [241] proposes a joint learning framework for accurate and explainable recommendations by integrating induction of several explainable rules from item association, such as, *Also view, Buy after view, Also buy, and Buy together*, where such explanations can sufficiently affect users online behaviors in e-commerce platforms.

5.3.2 Healthcare. Recently, AI techniques have demonstrated their significant impact on healthcare (e.g., drug discovery, disease diagnosis, gene analysis, etc.) in which decisions need to be carefully made based on solid evidence. For example, explainable AI has been proven to significantly accelerate the process of computer-assisted drug discovery [189, 344], such as molecular design, chemical synthesis planning, protein structure prediction, and macromolecular target identification. Explanations of graph neural networks have been conducted on a set of molecules graph-labeled for their mutagenic effect on the Gram-negative bacterium Salmonella typhimurium, with the goal of identifying several known mutagenic functional groups $\text{NH}_2$ and $\text{NO}_2$ [240, 372, 378]. A recent work [282] studies how the interpretation of filters within message-passing networks can lead to the identification of relevant toxicophore- and pharmacophore-like sub-structures for explainability, so as to help increase their reliability and foster their acceptance and usage in drug discovery and medicinal chemistry projects in the healthcare domain. In disease diagnosis, When AI models meet doctors who are experts in their sector for disease diagnosis, explanations should be made by considering the doctor’s knowledge, application domain, the disease, patient conditions, etc. What’s more, explanations should give doctors confidence in AI models’ prediction that can lead to higher quality service [16].

5.3.3 Natural Language Processing. As one of the most broadly applied areas of AI, Natural Language Processing (NLP) investigates the use of computers to process or understand human (i.e., natural) languages [114]. Applications of NLP are everywhere, including dialogue systems, text summarization, machine translation, question answering, sentiment analysis, information retrieval, etc. Recently, deep learning approaches have obtained very promising performance across many different NLP tasks, which comes at the expense of models becoming less explainable [110, 265]. As a consequence, these NLP systems cannot be fully trusted, and humans might doubt how particular decisions are made. To address the issue, LIME [289] proposes to generate random input perturbations for a given document to explain the predicted categories for text classification in SVM models. CAML [265] employs an attention mechanism to select the segments that are most relevant for medical codes (ICD) from clinical text.

5.4 Surveys and Tools

In this subsection, we introduce existing surveys, tools and repositories on explainability in AI to facilitate the readers who wish to further explore this field.
5.4.1 Surveys. In the book [262], the author focuses on interpretable machine learning by introducing fundamental concepts to advanced interpretable models. For example, it first details related concepts of interpretability, followed by intrinsically interpretable models, such as linear regression, decision tree, rule-based methods, etc. Afterward, the book provides general post-hoc tools for interpreting black-box models and explaining individual predictions. Doshi-Velez et al. [120] raises the importance of intractability in machine learning and introduces a comprehensive survey at this field. There are surveys [29, 37, 121, 153, 163, 227] summarizing explanation approaches in machine learning. In addition, comprehensive surveys for specific applications also exist, such as recommender systems [391], medical information systems [338], natural language processing [110], graph neural networks [379], drug discovery [189], etc.

5.4.2 Tools. In this subsection, we introduce several popular toolkits that are open-sourced in the GitHub platform for explainable AI. AIX360\(^3\) (AI Explainability 360) [30] is an open-source Python toolkit featuring state-of-the-art explainability methods and some evaluation metrics. Meanwhile, AIX360 also provides educational materials for non-technical stakeholders to quickly become familiar with interpretation and explanation methods. InterpretML\(^4\) [270] is also an open-source python toolkit that exposes machine learning interpretability algorithms to practitioners and researchers. InterpretML exposes two types of interpretability: glass-box for machine learning models with model-intrinsic explanations, and black-box explainability techniques for explaining any existing AI systems. The package DeepExplain\(^5\) [27] mainly supports various gradient-based techniques and perturbation-based methods. In addition, DIG\(^6\) [233] provides python toolkit for explaining graph deep learning. A summary of the above-mentioned open-source toolkits and frameworks on explainable AI can be found in Table 7.

<table>
<thead>
<tr>
<th>Toolkit</th>
<th>Characteristics</th>
</tr>
</thead>
<tbody>
<tr>
<td>InterpretML(^7) [270]</td>
<td>A python toolkit for supporting intrinsically transparent models and black-box models.</td>
</tr>
<tr>
<td>AIX360(^8) [30]</td>
<td>A python toolkit with comprehensive set of explainability methods and datasets.</td>
</tr>
<tr>
<td>DeepExplain(^9) [27]</td>
<td>A Tensorflow as well as Keras with Tensorflow backend toolkit for mainly supporting various gradient-based techniques and perturbation-based methods.</td>
</tr>
</tbody>
</table>

5.5 Future Directions

In this subsection, we discuss potential directions for future research in explainable AI. Since the interpretability of AI is a relatively new and still a developing area, many open problems need to be considered.

- **Security of explainable AI.** Recent studies have demonstrated that due to their data-driven nature, explanations of AI models are vulnerable to malicious manipulations. Attackers attempt to generate adversarial examples that can not only mislead a target classifier but also deceive its corresponding interpreter [152, 390], naturally raising potential security concerns on interpretations. Therefore, learning how to defend against adversarial attacks on explanations would be an important future direction for research.

- **Evaluation Methodologies.** Evaluation metrics are crucial for studying explanation methods; however, due to the lack of ground truths and human subjective understandings, evaluating whether the explanations are

---

\(^3\)https://aix360.mybluemix.net
\(^4\)https://github.com/interpretml/interpret
\(^5\)https://github.com/marcoancona/DeepExplain
\(^6\)https://github.com/divelab/DIG
reasonable and correct in regard to certain predictions is becoming intractable. The widely used evaluation methodology is based on human evaluations based on visualizing explanations, which is time-consuming and biased toward subjective human understandings. Although there are some initial studies on the evaluation of interpretability [120], it is still unclear how to measure what constitutes a good Explanation?”. It is crucial to investigate qualitative and quantitative evaluations of interpretability.

- **Knowledge to Target model: from white-box to black-box.** Most existing explanation techniques require full knowledge of the explained AI system (denoted as white-box). However, knowledge regarding target AI systems is often limited in many scenarios due to privacy and security concerns. Therefore, an important direction is to understand how an explanation can be generated for making decisions in black-box systems.

## 6 PRIVACY

The success of modern AI systems is built upon data, and data might contain private and sensitive information – from credit card data to medical records and from social relations to family trees. To establish trustworthy AI systems, the safety of private and sensitive information carried by the data and models that could be potentially exposed throughout the AI system must be guaranteed. Therefore, increasing attention has been paid to the protection and regulation of data privacy. From a legal perspective, laws from the state level to the global level have begun to provide mandatory regulations for data privacy. For instance, the California Consumer Privacy Act (CCPA) was signed into law in 2018 to enhance privacy rights and consumer protection in California by giving consumers more control over the personal information that businesses collect; the Health Insurance Portability and Accountability Act (HIPAA) was created in 1996 to protect individual healthcare information by requiring authorization before disclosing personal healthcare information; the European Union announced General Data Protection Regulation (GDPR) to protect data privacy by giving the individual control over personal data collection and usage.

From the perspective of science and technology, although most AI technologies haven’t considered privacy as the fundamental merit when they are first developed, to make modern AI systems trustworthy in privacy protection, a subfield of AI, privacy-preserving machine learning (PPML), has set privacy protection as the priority and has begun to pioneer principled approaches for preserving privacy in machine learning. Specifically, researchers uncover the vulnerabilities of existing AI systems from comprehensive studies and then develop promising technologies to mitigate these vulnerabilities. In this section, we will provide a summary of this promising and important field. Specifically, the basic concepts and taxonomy will be first discussed, and the risk of privacy breaches will be explained through various privacy attacking methods. Mainstream privacy-preserving technologies, such as confidential computing, federated learning, and differential privacy will be included, followed by discussions on applications in real systems, existing surveys and tools, and the future directions.

### 6.1 Concepts and Taxonomy

In the context of privacy protection, the adversarial goal of an attacker is to extract information about the data or machine learning models. According to the accessible information the adversary has, the attacker can be categorized into **white-box or black-box**. In a white-box setting, we assume that the attacker has all information except the data that we try to protect and the attacker aims to attack. In a black-box setting, the attacker has very limited information, for example, the query results returned by the model. Based on when the attack occurs, the privacy breach could happen in the **training phase or inference phase**. In the training phase, the adversary might be able to directly access or infer the information about the training data when she inspects or even tampers with the training process. In the inference
phase, the adversary might infer the input data of the model by inspecting the output characteristics. According to the capability of the adversary, the attacker may be honest-but-curious or fully malicious. An honest-but-curious attacker can inspect and monitor the training process while a fully malicious attacker can further tamper the training process. These taxonomies are not exclusive since they view the attacker from different perspectives.

6.2 Methods
We will highlight the risk of privacy leakage by introducing some representative privacy attack methods. Then, some mainstream techniques for privacy-preserving will be introduced.

6.2.1 Privacy Attack. Privacy attacks can target training data, input data, properties of data population, and even the machine learning model itself. We introduce some representative privacy attacks to reveal the risk of privacy breaches.

Membership Inference Attack. To investigate how machine learning models leak information about individual data within the training data, the membership inference attack aims to identify whether a data record is used in the training of machine learning models. For instance, given the black-box access to the model, an inference model can be trained to recognize whether the given inputs are used in its training or not based on the predictions of the target model [321]. Empirically, it is shown that commonly used classification models can be vulnerable to membership inference attacks. Therefore, private information can be inferred if some user data (e.g., medical records and credit card data) is used in training the model. Please refer to the survey [179] for a comprehensive summary of membership inference attacks.

Model Inversion Attack. Model inversion attack [145, 146] aims to use the model’s output to infer the information of the input data that often contain sensitive and private information. For instance, in pharmacogenetics, machine learning models are used to guide medical treatments, given the patient’s genotype and demographic information. However, it has been shown that severe privacy risk exists because a patient’s genetic information can be disclosed given the model and the patient’s demographic information [146]. In facial recognition with neural networks, the images of people’s faces can be recovered given their names, prediction confidence values, and access to the model [145]. In [392], generative adversarial networks (GANs) are used to guide the inversion process of neural networks and reconstruct high-quality facial images from face recognition classifiers. In a recent study, researchers found that the input data can be perfectly recovered through the gradient information of neural networks [401], which highlights the privacy risk in distributed learning where gradient information needs to be transmitted when people used to believe that it can preserve data privacy.

Property Inference Attack. Given the machine learning model, the property inference attack aims to extract global properties of the training dataset or training algorithm that the machine learning models do not intend to share. One example is to infer the properties that only hold for a subset of the training data or a specific class of the training data. This type of attack might leak private statistical information about the population, and the learned property can be used to exploit the vulnerability of an AI system.

Model Extraction. An adversary aims to extract the model information by querying the machine learning model in a black-box setting such that he can potentially fully reconstruct the model or create a substitute model that closely approximates the target model [341]. Once the model has been extracted, the black-box setting translates to the white-box setting, where other types of privacy attacks become much easier. Moreover, the model information typically contains an intelligent property that should be kept confidential. For instance, ML-as-a-service (MLaaS) systems, such as Amazon AWS Machine Learning, Microsoft Azure Machine Learning Studio, and Google Cloud Machine Learning
Engine, allow users to train the models on their data and provide publicly accessible query interfaces on a pay-per-query basis. The confidential model contains users’ intelligent property but suffers from the risk of functionality stealing. Since the trained ML model is often considered as an intellectual property (IP), researchers recently have proposed IP protection solutions to protect ML models from infringement. The taxonomy, attacks, and evaluations in this research direction are summarized in a recent survey [367].

6.2.2 Privacy Preservation. The privacy-preserving countermeasures can be roughly categorized into three mainstream and promising directions, including confidential computing, federated learning, and differential privacy, as shown in Figure 11. Confidential computing attempts to ensure data safety during transmission and computing. Federated learning provides a new machine learning framework that allows data to be local and decentralized and avoid raw data transmission. Differential privacy aims to utilize the information about a whole dataset without exposing individual information in the dataset. Next, we review these techniques and discuss how they preserve privacy.

![Fig. 11. An Overview of Privacy Preserving Techniques](image)

**Confidential Computing.** There are mainly three types of techniques for achieving confidential computing, including Trusted Executive Environment (TEE) [304], Homomorphic Encryption (HE) [17], and Multi-party Secure Computation (MPC) [129].

**Trusted Execution Environments.** Trusted Execution Environments focus on developing hardware and software techniques to provide an environment that isolates data and programs from the operator system, virtual machine manager, and other privileged processes. The data is stored in the trusted execution environment (TEE) such that it is impossible to disclose or operate on the data from outside. The TEE guarantees that only authorized codes can access the protected data, and the TEE will deny the operation if the code is altered. As defined by the Confidential Computing Consortium [12], the TEE provides a level of assurance of data confidentiality, data integrity, and code integrity that essentially states that unauthorized entities cannot view, add, remove, or alter the data while it is in use within the TEE, and cannot add, remove or alter code executing in the TEE.

**Secure Multi-party Computation.** Secure multi-party computation (MPC) protocols aim to enable a group of data owners who might not trust one another to jointly perform a function computation that depends on all of their private input without disclosing any participant’s private data. Although the concept of secure computation was primarily a theoretical interest when it was first proposed [370], it has now become a practical tool to enable privacy-preserving applications in which multiple distrusting data owners seek to compute a function cooperatively [129].
**Homomorphic Encryption.** Homomorphic Encryption (HE) enables computation functions on the data without accessing the plaintext by allowing mathematical operations to be performed on ciphertext without decryption. It returns the computation result in the encrypted form, which can be decrypted just as the computation is performed on the decrypted data. With partially homomorphic encryption schemes, only certain operations can be performed, which limits them to specialized problems that can be reduced as the supported operations. Fully-homomorphic encryption (FHE) schemes aim to provide support for a universal set of operations so that any finite function can be computed. The first FHE scheme was proposed by Gentry [150], and was based on lattice-based cryptography. There have been a lot of recent interests in implementing FHE schemes [90, 151], but to build a secure, deployable, scalable system using FHE is still challenging.

**Federated Learning.** Federated learning (FL), as shown in Figure 12, is a popular machine learning paradigm where many clients, such as mobile devices or sensors, collaboratively train machine learning models under the coordination of a central server, while keeping the training data from the clients decentralized [250]. This paradigm is in contrast with traditional machine learning settings, where the data is first collected and transmitted to the central server for further processing. In federated learning, the machine learning models are moving between the server and clients while keeping the private data locally within the clients. Therefore, it essentially avoids the transmission of private data and significantly reduces the risk of privacy breaches.

Next, we briefly describe a typical workflow for a federated learning system [250]:

- **Client selection:** The server samples a subset of clients from those active clients according to some eligibility requirements.
- **Broadcast:** The server broadcasts the current model and the training program to the selected clients.
- **Local computation:** The selected clients locally compute the update to the received model based on the local private data. For instance, the stochastic gradient descent (SGD) update can be run with the stochastic gradient computed based on local data and the model.
- **Aggregation:** The server collects the updated local models from the selected clients and aggregates them as an updated global model.

This workflow represents one round of the federated learning algorithm, and it will repeat until reaching specific requirements, such as convergence accuracy or performance certificates.

In addition to protecting data privacy by keeping the data local, there are many other techniques to further secure data privacy. For instance, we can apply lossy compression before transferring the models between server and clients.
such that it is not easy for the adversary to infer accurate information from the model update [401]. We also can apply secure aggregation through secure multi-party computation such that no participant knows the local model information from the other participants, but the global model can still be computed [23, 261]. Additionally, we can also apply noisy perturbation to improve the differential privacy [351].

Federated learning is becoming an increasingly popular paradigm for privacy protection and has been studied, developed, and deployed in many applications. However, federated learning still faces many challenges, such as the efficiency and effectiveness of learning, especially with non-IID data distributions [203, 204, 223, 224, 234].

**Differential Privacy.** Differential Privacy (DP) is an area of research that aims to provide rigorous statistical guarantees for reducing the disclosure about individual information in a dataset [122, 125]. The major idea is to introduce some level of uncertainty through randomization or noise into the data such that the contribution of individual information is hidden while the algorithm can still leverage valuable information from the dataset as a whole. According to the definition [125], let’s first define that the datasets $D$ and $D'$ are adjacent if $D'$ can be obtained from $D$ by altering the record of a single individual. A randomized algorithm $A$ is $(\epsilon, \delta)$-differentially private if for all $S \subseteq \text{Range}(A)$ and for all adjacent datasets $D$ and $D'$ such that

$$
\Pr[A(D) \in S] \leq e^\epsilon \Pr(A(D') \in S) + \delta.
$$

$(\epsilon, \delta)$ quantifies how much information can be inferred about an individual from the output of the algorithm $A$ on the dataset. For instance, if $\epsilon$ and $\delta$ are sufficiently small, the output of the algorithm will be almost identical, i.e., $\Pr[A(D) \in S] \approx \Pr(A(D') \in S)$, such that it is difficult for the adversary to infer the information of any individual since the individual’s contribution on the output of the algorithm is nearly masked. The privacy loss incurred by the observation $\xi$ is defined as

$$
\mathcal{L}_{A,D,D'}^\xi = \ln \left( \frac{\Pr[A[D] = \xi]}{\Pr[A[D'] = \xi]} \right).
$$

$(\epsilon, \delta)$-differential privacy ensures that for all adjacent datasets $D$ and $D'$, the absolute value of the privacy loss is bounded by $\epsilon$ with probability at least $1 - \delta$. Some common methods to provide differential privacy include random response [350], Gaussian mechanism [125], Laplace mechanism [124], exponential mechanism [252], etc. Different from other techniques, differential privacy provides provable guarantees and measurements for privacy protection, but the challenge is to increase the utility under the given budget, especially for high-dimensional models.

### 6.3 Applications in Real Systems

Privacy-preserving techniques have been widely used to protect sensitive information in real systems. In this subsection, we discuss some representative examples.

#### 6.3.1 Healthcare

Healthcare data can be available from patients, clinical institutions, insurance companies, pharmacies, and so on. However, the privacy concern of personal healthcare information makes it difficult to fully exploit the large-scale and diverse healthcare data to develop effective predictive models for healthcare applications. Federated learning provides an effective privacy-preserving solution for such scenarios since data across the population can be utilized while not being shared [3, 196, 291, 319, 365, 383]. Differential privacy has also gained significant attention as a general way of protecting healthcare data [111].

#### 6.3.2 Biometric Data Analysis

Biometric data is mostly non-revocable and can be used for identification and authentication. Therefore, it is critical to protect private biometric data. To this end, confidential computing, federated learning,
and differential privacy techniques have become widely applied to protect people’s biometric data such as face images, medical images, and fingerprint pattern [53, 196, 305, 347].

6.3.3 **Recommender Systems.** Recommender systems utilize users’ interactions on products such as movies, music, and goods to provide relevant recommendations. The rating information has been shown to expose users to inference attacks, leaking private user attributes such as age, gender, etc [24, 251, 322]. To protect user privacy, recent works [202, 251, 267, 388] have developed privacy-preserving recommender systems via differential privacy.

6.3.4 **Distributed Learning.** In distributed learning, it is possible to recover a client’s original data from their gradient information or model update [395, 401]. Secure multiparty computing is applied to protect the information of locally trained models during aggregation [23, 261, 298]. The Differentially Private SGD [15, 186, 327] provides an effective and provable manner to protect the input data by adding noise in gradient and has been popular in the training of deep learning models.

For models with a large number of trainable parameters, the differentially private models degrade the utility drastically since the added noise suffers from dimensional dependency [33]. Various subspace methods [194, 374, 375, 400] have been proposed to circumvent the dependence on the ambient dimension by leveraging the empirical observation on the low-dimensional structure of gradient space in deep networks, and they have been shown to increase the utility significantly. It is shown in [340] that transfer learning of public data from a similar domain helps improve the utility of private learning. There also exist discussions on the interplay between memorization and privacy leakage in machine learning [54, 66, 140].

6.3.5 **Natural Language Processing.** Recent studies [67] that can extract private data from large language models such as GTP-2 have exacerbated the privacy concerns. To mitigate these concerns, differentially private SGD (DP-SGD) has been applied to the training of large-scale training of language models such as BERT [28, 225, 373], and many techniques and tricks have been proposed to mitigate the performance drop and improve the training efficiency. It is suggested that privately learning with a pre-trained model tends to not suffer from dimension-dependent performance degradation [225, 373].

6.4 **Surveys and Tools**

We collect some surveys and tools relevant to privacy in AI systems for further exploration.

6.4.1 **Surveys.** The general concepts, threats, attack, and defense methods in privacy-preserving machine learning are summarized in several surveys [26, 112, 292]. Federated learning is comprehensively introduced in the papers [250, 368]. Differential privacy is reviewed in the surveys [122, 125, 187].

6.4.2 **Tools.** Popular tools and repositories in federated learning include TensorFlow Federated (TFF) [13], FATE [5], FedML [171], PaddleFL [11] and LEAF [6]. Popular tools in differential privacy include Facebook Opacus [10], TensorFlow-Privacy [14], OpenDP [9] and Diffpriv [299]. Keystone Enclave [218] is an open framework for designing Trusted Execution Environments. Popular tools in Secure Multiparty Computing and Homomorphic Encryption are summarized in the lists [7, 8]. These tools are also summarized in Table 8.
Table 8. Representative open-source toolkits and frameworks on private AI.

<table>
<thead>
<tr>
<th>Toolkit</th>
<th>Characteristics</th>
</tr>
</thead>
<tbody>
<tr>
<td>FedML [171]</td>
<td>A versatile Edge-Cloud Ecosystem for federated learning and analytics at scale.</td>
</tr>
<tr>
<td>LEAF [6]</td>
<td>A benchmarking framework for federated learning, with applications including federated learning, multi-task learning, meta-learning, and on-device learning.</td>
</tr>
</tbody>
</table>

6.5 Future Directions

Confidential computing, federated learning, and differential privacy are three effective ways to improve privacy protection. However, they are far away from being extensively used and require more development. For instance, the computation efficiency and flexibility of confidential computing are not mature enough to support the applications of AI systems in our society. There are also great challenges to improve the efficiency and effectiveness of federated learning when deploying in large-scale and heterogeneous environments. It would be desirable to achieve a better trade-off between utility and privacy loss in differential privacy. Most importantly, a versatile and reliable system design for achieving privacy protection and different techniques should be integrated to enhance the trustworthiness of AI systems by reducing the risks of privacy leakages from multiple perspectives.

7 ACCOUNTABILITY & AUDITABILITY

In general, accountability for AI indicates how much we can trust these AI technologies and who or what we should blame if any parts of the AI technologies perform below expectation. It is about a declaration of responsibility. It is not trivial to explicitly determine the accountability for AI. On the one hand, most AI-based systems act as “black-box”, due to the lack of explainability and transparency. On the other hand, real-world AI-based systems are very complex, and involve numerous key components, including input data, algorithm theory, implementation details, real-time human control, and so on. These factors further complicate the determination of accountability for AI. Although difficult and complex, it is necessary to guarantee accountability for AI. Auditability, which refers to a set of principled evaluations of the algorithm theories and implementation processes, is one of the most important methodologies in guaranteeing accountability.

It is very important to achieve a balance between accountability and innovation in AI. The overall aim is for humans to enjoy the benefits and conveniences of AI with a reliable and guarantee of safety. Additionally, however, we do not want to heavily burden the algorithm designer or put too many restrictions on end-users of AI-based systems. In this section, we discuss the accountability and auditability of AI. First, we introduce the basic concept of accountability and some key roles within it. We then describe the definition of auditability for AI and two kinds of audits. Finally, we
summarize existing surveys and tools, and discuss some future directions to enhance accountability and auditability in AI.

7.1 Concepts and Taxonomy

In this subsection, we will introduce the key concepts and taxonomies of accountability and auditability in AI.

7.1.1 Accountability. Accountability in AI has a broad definition. On the one hand, accountability can be interpreted as a property of AI. From this perspective, accountability can be improved if breakthroughs can be made in the explainability of AI algorithms. On the other hand, accountability can be referred to as a clear responsibility distribution, which focuses on who should take the responsibility for each impact of AI-based systems. Here we mainly focus on discussing the second notion. As indicated above, it is not trivial to give a clear specification for responsibility, since the operation of an AI-based system involves many different parties, such as the system designer, the system deployer, and the end-user. Any improper operation from any parties may result in system failure or potential risk. Also, all kinds of possible cases should be taken into consideration to ensure a fair distribution of responsibility. For example, the cases when an AI system does harm when working correctly versus working incorrectly should be considered differently [245, 377]. To better specify accountability, it is necessary to determine the roles and the corresponding responsibility of different parties in the function of an AI system. In [353], three roles are proposed: decision-makers, developers, and users. By refining these three roles, we propose five roles, and introduce their responsibilities and obligations as follows:

- **System Designers:** system designers are the designers of the AI system. They are supposed to design an AI system that meets the user requirements and is transparent and explainable to the greatest extent. It is their responsibility to offer deployment instructions and user guidelines, and to release potential risks.
- **Decision Makers:** decision-makers have the right to determine whether to build an AI system and what AI system should be adopted. Decision-makers should be fully aware of the benefits and risks of the candidate AI system, and take all the relevant requirements and regulations into overall consideration.
- **System Deployers:** system deployers are in charge of deploying an AI system. They should follow the deployment instructions carefully and ensure that the system has been deployed appropriately. Also, they are expected to offer some hands-on tutorials to the end-users.
- **System Auditors:** system auditors are responsible for system auditing. They are expected to provide comprehensive and objective assessments of the AI system.
- **End Users:** end-users are the practical operators of an AI system. They are supposed to follow the user guidelines carefully and report emerging issues to system deployers and system designers in a timely fashion.

7.1.2 Auditability. Auditability is one of the most important methodologies in ensuring accountability, which refers to a set of principled assessments from various aspects. In the IEEE standard for software development [1], an audit is defined as “an independent evaluation of conformance of software products and processes to applicable regulations, standards, guidelines, plans, specifications, and procedures.” Typically, audits can be divided into two categories as follows:

- **External audits:** external audits [161, 308] refer to audits conducted by a third party that is independent of system designers and system deployers. External audits are expected to share no common interest with the internal workers and are likely to provide some novel perspectives for auditing the AI system. Therefore, it is expected that external audits can offer a comprehensive and objective audit report. However, there are obvious limitations to external audits. First, external audits typically cannot access all the important internal data in an AI system, such as the model training data and model implementation details [60], which increases the auditing difficulty. Additionally, external audits are
always conducted after an AI system is deployed, so that it may be costly to make adjustments over the system, and, sometimes, the system may have already done harm [264].

Internal audits: internal audits refer to audits conducted by a group of people inside the system designer or system deployer organizations. SMACTR [287] is a recent internal auditing framework proposed by researchers from Google and Partnership on AI and consists of five stages: scoping, mapping, artifact collection, testing, and reflection. Compared with external audits, internal audits can have access to a large amount of internal data, including the model training data and model implementation details, which makes internal audits much more convenient. Furthermore, internal audits can be conducted before an AI system is deployed, thus avoiding some potential harm after the system’s deployment. The internal audit report can also serve as an important reference for the decision-maker to make a decision. However, an unavoidable shortcoming for internal audits is that they share the same interest as the audited party, which makes it challenging to give an objective audit report.

7.2 Surveys and Tools
In this subsection, we summarize existing surveys and tools about accountability and auditability of AI, to facilitate readers who want to explore this field further.

7.2.1 Surveys. A recent work on algorithmic accountability is presented in [353]. It takes Boven’s definition of accountability [52] as the basic concept and combines it with numerous literature in algorithmic accountability to build the concept’s definition.

7.2.2 Tools. The other five dimensions (safety & robustness, non-discrimination & fairness, explainability, privacy, environmental well-being) discussed in this survey are also important aspects to be evaluated during algorithm auditing. Therefore, most tools introduced in section 3.6, 4.4.2, 5.4.2, 6.4, and 8.3 can also be used for the purpose of auditing.

7.3 Future Directions
For accountability, it is important to further enhance the explainability of the AI system. Only when we have a deep and thorough understanding of its theory and mechanism can we fully rely on it or make a well-recognized responsibility distribution scheme. For auditability, it is always a good option to conduct both external audits and internal audits, so that we can have a comprehensive and objective overview of an AI system. Furthermore, we need to be aware that an AI system is constantly dynamic. It can change with input data and environment. Thus, to make an effective and timely audit, it is necessary to audit the system periodically and to update auditing principles with the system changes [228].

8 ENVIRONMENTAL WELL-BEING
A trustworthy AI system should be sustainable and environmentally friendly [325]. In fact, the large-scale development and deployment of AI systems bring a huge burden of energy consumption, which inevitably affects the environment. For example, Table 9 shows the carbon emission (as an indicator of energy consumption) of training NLP models and that of daily consumption [330]. We find that training a common NLP pipeline has the same carbon emissions as a human produces in seven years. Training and fine-tuning a large Transformer model costs five times more energy consumption than a car over its lifetime. Besides model development, in other areas, such as data center cooling, there is also a huge energy cost. The rapid development of AI technology further challenges the tense global situation

of energy shortage and environmental deterioration. Hence, environmental friendliness becomes an important issue to consider in building trustworthy AI systems. In this section, we review the existing works regarding the environmental impacts of AI technologies. Existing works mainly focus on the impact of AI systems’ energy consumption on the environment. We first present an overview of the strategies for reducing energy consumption, e.g., model compression, and then introduce the works on estimating the energy consumption and evaluating the environmental impacts of real-world AI systems in different domains. Finally, we summarize the existing surveys and tools on this dimension.

<table>
<thead>
<tr>
<th>Consumption</th>
<th>CO₂e (lbs)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Air travel, 1 passenger, NY↔SF</td>
<td>1984</td>
</tr>
<tr>
<td>Human life, avg. 1 year</td>
<td>11,023</td>
</tr>
<tr>
<td>American life, avg. 1 year</td>
<td>36,156</td>
</tr>
<tr>
<td>Car, avg incl. fuel, 1 lifetime</td>
<td>126,000</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Training one model (GPU)</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>NLP pipeline (parsing, SRL) w/ tuning &amp; experimentation</td>
<td>78,468</td>
</tr>
<tr>
<td>Transformer (big) w/ neural architecture search</td>
<td>626,155</td>
</tr>
</tbody>
</table>

Table 9. Comparisons between estimated CO₂ emissions produced by daily lives and training NLP models. (Table Credit: [330])

8.1 Methods

In this subsection, we summarize the techniques developed for reducing the energy use of AI algorithms. Improving the energy efficiency of AI systems involves algorithm-level and hardware-level solutions. We will introduce two common classes of algorithm-level approaches: model compression and adaptive design, as well as hardware-level energy-saving methods.

8.1.1 Model Compression. Model compression is a hot topic in deep learning that receives continuous attention from both academia and industry. It studies how to reduce the size of a deep model to save storage space and the energy consumption for training and deploying models, with an acceptable sacrifice on model performance. For CNN models in the image domain, parameter pruning and quantization [92, 167], low-rank factorization [185, 293], transferred/compact convolutional filters [97, 318], and knowledge distillation have been proposed [176, 296]. Similarly, in the text domain, researchers borrow and extend these methods: pruning [65, 256], quantization [88, 177], knowledge distillation [207, 331, 334], and parameter sharing [113, 216], to compress popular NLP models, such as Transformer and BERT.

8.1.2 Adaptive Design. Another line of research focuses on adaptively designing a model architecture to optimize the energy efficiency of a model. Yang et al. [369] propose a pruning approach to design CNN architectures to achieve an energy-saving goal. In their method, the model is pruned in a layer-by-layer manner, where the layer that consumes the most energy is pruned first. Stamoulis et al. [328] propose a framework to adaptively design CNN models for image classification under energy consumption restrictions. They formulate the design of a CNN architecture as a hyperparameter optimization problem and solve it by Bayesian optimization.
8.1.3 Hardware. In addition to the algorithm level, endeavors are also conducted to improve the energy efficiency of AI from the design of the hardware. Computing devices or platforms specially designed for AI applications are proposed to maximize the training and inference efficiency of AI algorithms. Specifically, hardware designed for DNN models are called DNN accelerators [85]. Esmaeilzadeh et al. [127] design a neural processing unit (NPU) to execute the fixed computations of a neuron such as multiplication, accumulation, and sigmoid, on chips. Later, Liu et al. [235] proposed RENO, which is a more advanced on-chip architecture for neural network acceleration. There is also hardware designed for specific NN structures. Han et al. [166] investigate how to design an efficient computation device for a sparse neural network, where weight matrices and feature maps are sparse. They [165] also devise an efficient speech recognition engine that is dedicated to RNN models. Furthermore, ReGAN [77] is developed to accelerate generative adversarial networks (GANs).

8.2 Applications in Real Systems

As described before, the environmental impacts of AI systems mainly come from energy consumption. In this subsection, we introduce the research on evaluating and estimating the energy consumption of real-world AI systems in different domains.

In the field of computer vision, Li et al. [222] first investigate the energy use of CNNs on image classification tasks. They provide a detailed comparison among different types of CNN layers, and also analyze the impact of hardware on energy consumption. Cai et al. [61] introduce the framework NeuralPower which can estimate the power and runtime across different layers in a CNN, to help developers to understand the energy efficiency of their models before deployment. They also propose evaluating CNN models with a novel metric “energy-precision ratio”. Based on it, developers can trade off energy consumption and model performance according to their own needs, and choose the appropriate CNN architecture. In the field of NLP, Strubell et al. [330] examine the carbon emissions of training popular NLP models, namely, Transformer, ELMo, and GPT-2, on different types of hardware, and shed light on the potential environmental impacts of NLP research and applications.

8.3 Surveys and Tools

In this subsection, we collect related surveys and tools on the dimension of environmental well-being.

8.3.1 Surveys. From the algorithm-level perspective, García-Martín et al. [147] present a comprehensive survey on energy consumption estimation methods from both the computer architecture and machine learning communities. Mainly, they provide a taxonomy for the works in computer architecture and analyze the strengths and weaknesses of the methods in various categories. Cheng et al. [87] summarize the common model compression techniques and organize them into four categories, and then present a detailed analysis on the performance, application scenarios, advantages, and disadvantages of each category. In the hardware-level perspective, Wang et al. [348] compares the performance and energy consumption of the processors from different vendors for AI training. Mittal and Vetter [259] review the approaches for analyzing and improving GPU energy efficiency. The survey [85] summarizes the latest progress on DNN accelerator design.

8.3.2 Tools. SyNERGY [295] is a framework integrated with Caffe for measuring and predicting the energy consumption of CNNs. Lacoste et al. [214] develop a Machine Learning Emissions Calculator as a tool to quantitatively estimate the carbon emissions of training an ML model, which can enable researchers and practitioners to better understand the environmental impact caused by their models. Accelergy [361] and Timeloop [276] are two representative energy
estimation tools for DNN accelerators. A summary of representative open-source toolkits on environmental well-being in AI can be found in Table 10.

Table 10. Representative open-source toolkits and frameworks on environmental well-being in AI.

<table>
<thead>
<tr>
<th>Toolkit</th>
<th>Characteristics</th>
</tr>
</thead>
<tbody>
<tr>
<td>MLEC [214]</td>
<td>A tool for estimating the carbon emissions of training a ML model.</td>
</tr>
<tr>
<td>Timeloop [276]</td>
<td>A systematic DNN accelerator evaluation tool that supports energy estimation.</td>
</tr>
</tbody>
</table>

8.4 Future Directions

Research on reducing the energy consumption of AI systems for environmental well-being is on the rise. At the algorithmic level, automated machine learning (AutoML), which aims to automatically design effective and efficient model architectures for certain tasks, emerges as a novel direction in the AI community. Existing works in AutoML focus more on designing an algorithm to improve its performance, but don’t usually treat energy consumption savings as the highest priority. Using AutoML technologies to design energy-saving models needs further exploration in the future. At the hardware level, current research on DNN accelerators pays more attention to devising efficient deployment devices to facilitate model inference, but the procedure of model training is overlooked. The design of efficient customized training devices for various DNN models is a practical and promising direction to investigate in the future.

9 INTERACTIONS AMONG DIFFERENT DIMENSIONS

An ideal trustworthy AI system should simultaneously satisfy the six dimensions discussed above. In reality, the six dimensions are not independent of one another. The satisfaction of one dimension can promote the pursuit of another dimension. Meanwhile, conflicts exist among different dimensions. The realization of one dimension could violate another dimension, which makes it impossible for two or more dimensions to be met simultaneously in some scenarios. Researchers and practitioners should be aware of the complicated interactions among different dimensions. Knowing the accordance between two dimensions brings us an alternative idea to achieve one dimension: we can try to satisfy one dimension by realizing the other. Moreover, when two dimensions are contradictory, we can make a trade-off between them according to our needs. In this section, we discuss some known accordance and conflict interactions among different dimensions.

9.1 Accordance

Two dimensions are accordant when the satisfaction of one dimension can facilitate the achievement of the other, or the two dimensions promote each other. Next, we show two examples of accordance interactions among dimensions.

Robustness & Explainability. Studies show that deep learning models’ robustness against adversarial attacks positively correlates with their explainability [128, 133, 269]. Etmann et al. [128] find that models trained with robustness objectives show more interpretable saliency maps. Specifically, they prove rigorously in mathematics that Lipschitz regularization, which is commonly used for robust training, forces the gradients to align with the inputs. Noack et al. [269] further investigate the opposite problem: will an interpretable model be more robust? They propose Interpretation Regularization (IR) to train models with explainable gradients and empirically show that a model can be more robust to adversarial attacks if it is trained to produce explainable gradients.
**Fairness & Environmental Well-being.** Fairness in the field of AI is a broad topic, which involves not only the fairness of AI service providers and users, but also the equality of AI researchers. As mentioned in section 8, the development trend of deep learning models toward larger models and more computing resource consumption not only causes adverse environmental impact but also aggravates the inequality of research [330], since most researchers cannot afford high-performance computing devices. Hence, the efforts for ensuring the environmental well-being of AI techniques, such as reducing the cost of training large AI models, are in accordance with the fairness principle of trustworthy AI.

### 9.2 Conflict

Two dimensions are conflicting when the satisfaction of one dimension hinders the realization of the other. Next, we show three examples of the conflicting interactions among dimensions.

**Robustness & Privacy.** Recent studies find tensions between the robustness and the privacy requirements of trustworthy AI. Song et al. [326] check how the robust training against adversarial attacks influences the risk of a model against membership inference attack. They find that models trained with adversarial defense approaches are more likely to expose sensitive information in training data via membership inference attacks. The reason behind this is that models trained to be robust to adversarial examples typically overfit to training data, which makes training data easier to be detected from models’ outputs. From the perspective of the victims of unauthorized AI applications, the conflict between robustness and privacy is not always bad. Many large-scale AI applications, such as face recognition systems, are trained on data collected from people without their explicit consent, which raises concerns about the disclosure of personal information [280]. Thus, recent studies propose to take advantage of the defects in robustness of unauthorized AI models to prevent them from exploiting private data [180, 317, 335]. Specifically, Shan et al. [317] introduce imperceptible noise into users’ photos, so that it is hard for anyone who collects such images to train a face recognition model to correctly identify the users. Huang et al. [180] propose to add an error-minimizing noise into private training examples to fool the deep learning models into “believing there is nothing to learn from these examples”. It is worth mentioning that such poisoning attacks cannot solve the privacy issue once and for all. Models trained adaptively against the perturbed examples or new technologies developed after the attacks still have potential for nullifying the attacks [285].

**Robustness & Fairness.** Robustness and fairness can also conflict with each other in particular scenarios. As discussed in section 3, adversarial training is one of the mainstream approaches for improving the robustness of a deep learning model. Recent research [363] indicates that adversarial training can introduce a significant disparity of performance and robustness among different groups, even if the datasets are balanced. Thus, the adversarial training algorithm improves the robustness of a model at the expense of its fairness. Accordingly, the work [363] proposes a framework called Fair-Robust-Learning (FRL) to ensure fairness while improving a model’s robustness.

**Fairness & Privacy.** Cummings et al. [105] investigate the compatibility of fairness and privacy of classification models, and theoretically prove that differential privacy and exact fairness in terms of equal opportunity are unlikely to be achieved simultaneously. By relaxing the condition, this work further shows that it is possible to find a classifier that satisfies both differential privacy and approximate fairness.

**Addressing the Conflicts among Dimensions.** As discussed in this subsection, different dimensions of trustworthy AI can interact with one another in a conflicting manner. The conflicts make different dimensions restrict each other so sometimes we cannot meet them simultaneously to build a completely trustworthy AI system. Thus, how to deal with the conflicts among dimensions becomes a future research direction that draws increasing attention. There are two
potential directions. First, a direct problem to be solved is how to alleviate the conflict so as to meet both sides to the greatest extent [363]. Second, if the tension is proven to be unsolvable, more research on this field is needed for us to better understand how one dimension impacts one another, and how we can sacrifice one for another based on our actual needs [105].

10 FUTURE DIRECTIONS
In this survey, we elaborate on six of the most concerning and crucial dimensions an AI system needs to meet to be trustworthy. Beyond that, some dimensions have not received extensive attention, but are worth exploring in the future. In this section, we will discuss several other potential dimensions of trustworthy AI.

10.1 Human agency and oversight
The ethical guidelines for trustworthy AI proposed by different countries and regions all emphasize the human autonomy principle of AI technology [325]. Human autonomy prohibits AI agents from subordinating, coercing, or manipulating humans, and requires humans to maintain self-determination over themselves. To achieve the principle of human autonomy, the design of AI systems should be human-centered. Specifically, human agency and oversight should be guaranteed in the development and deployment of AI systems. Human agency enables humans to make decisions independently based on the outputs of an AI system, instead of being totally subject to AI’s decisions. A desirable human agency technology encourages users to understand the mechanism of an AI system and enables users to evaluate and challenge the decisions of an AI system, and make better choices by themselves. Human oversight enables humans to oversee AI systems throughout their life cycle, from design to usage. It can be achieved through human-in-the-loop, human-on-the-loop, and human-in-command governance strategies.

10.2 Creditability
With the wide deployment of AI systems, people increasingly rely on content produced or screened by AI, such as an answer to a question given by a question-answering (QA) agent or a piece of news delivered by a recommender system. However, the integrity of such content is not always guaranteed. For example, an AI system that exposes users to misinformation should not be considered trustworthy. Hence, additional mechanisms and approaches should be incorporated in AI systems to ensure their creditability.

10.3 Interactions among Different Dimensions
The research on the interactions among different dimensions is still in an early stage. Besides the several instances shown in this paper, there are potential interactions between other dimension pairs remaining to be investigated. For example, people may be interested in the relationship between fairness and interpretability. In addition, the interaction formed between two dimensions can be different in different scenarios, which needs further exploration. For example, an interpretable model may promote its fairness by making its decision process transparent. On the contrary, techniques to improve the interpretability of a model may introduce a disparity of interpretability among different groups, which leads to a fairness problem. Although there are numerous problems to study, understanding the interactions among different dimensions is very important in building a trustworthy AI system.
11 CONCLUSION

In this survey, we present a comprehensive overview of trustworthy AI from a computational perspective and clarify the definition of trustworthy AI from multiple perspectives, distinguishing it from similar concepts. We introduce six of the most crucial dimensions that make an AI system trustworthy; namely, Safety & Robustness, Nondiscrimination & Fairness, Explainability, Accountability & Auditability, Privacy, and Environmental Well-being. For each dimension, we present an overview of related concepts and a taxonomy to help readers understand, how each dimension is studied, and summarize the representative technologies, to enable readers to follow the latest research progress in each dimension.

To further deepen the understanding of each dimension, we provide numerous examples of applications in real-world systems and summarize existing related surveys and tools. We also discuss potential future research directions within each dimension. We then analyze the accordance and conflicting interactions among different dimensions. Finally, it is important to mention that outside of the six dimensions elaborated in this survey, there still exist some other potential issues that may undermine our trust in AI systems. Hence, we discuss several possible dimensions of trustworthy AI as future research directions.

REFERENCES


Weizhi Ma, Min Zhang, Yue Cao, Woojeong Jin, Chenyang Wang, Yiqun Liu, Shaoping Ma, and Xiang Ren. 2019. Jointly learning explainable rules.


Trustworthy AI: A Computational Perspective

Woodstock ’18, June 03–05, 2018, Woodstock, NY


[341] Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry. 2018. Robustness may be at odds with accuracy.

[342] Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry. 2018. Robustness may be at odds with accuracy.


