International Journal of Information Security (2024) 23:2287–2314
https://doi.org/10.1007/s10207-024-00813-3

REGULAR CONTRIBUTION

Trustworthy machine learning in the context of security and privacy

Ramesh Upreti[1,2] · Pedro G. Lind[1,2] · Ahmed Elmokashfi[3] · Anis Yazidi[1,2]

Published online: 3 April 2024

Abstract
Artificial intelligence-based algorithms are widely adopted in critical applications such as healthcare and autonomous vehicles. Mitigating the security and privacy issues of AI models, and enhancing their trustworthiness have become of paramount importance. We present a detailed investigation of existing security, privacy, and defense techniques and strategies to make machine learning more secure and trustworthy. We focus on the new paradigm of machine learning called federated learning, where one aims to develop machine learning models involving different partners (data sources) that do not need to share data and information with each other. In particular, we discuss how federated learning bridges security and privacy, how it guarantees privacy requirements of AI applications, and then highlight challenges that need to be addressed in the future. Finally, after having surveyed the high-level concepts of trustworthy AI and its different components and identifying present research trends addressing security, privacy, and trustworthiness separately, we discuss possible interconnections and dependencies between these three fields. All in all, we provide some insight to explain how AI researchers should focus on building a unified solution combining security, privacy, and trustworthy AI in the future.

Keywords Machine learning · Federated learning · Trustworthiness · Security · Privacy

1 Introduction and motivation

Development and investment in artificial intelligence (AI) technology is advancing at a rapid pace. AI has penetrated almost all life sectors from healthcare, and finance to space research. Despite the exponential adoption of AI-based solutions, several studies have unveiled some security and privacy vulnerabilities associated with AI systems [1–3]. In addition to this, some regulatory measures, namely the recent General Data Protection Regulation (GDPR), enforced by the European Union, the California Consumer Privacy Act (CCPA), enforced by the state of California in the USA and many other legislations introduced strong policies to ensure user data protection, preserve privacy and guarantee the security of data used in AI solutions.

Consequently, for proper regulation policies, the requirements of security and privacy-proof AI solutions have become of utmost importance and mandatory in today's AI world.

In traditional programming settings, the programmer knows how to generate output by creating rules or logic procedures working on input space. The success of the program (algorithm) is completely dependent on the ability of the programmer to write the code following the needed logic structure. This seems to be possible when the logic that maps the input to the output can be written using a sequence of conditional sentences (*if-then* statements). However, complex programming tasks such as face recognition involve rules and logic procedures that are impossible for humans to code because of two main issues. First, the complexity of the logic behind the programming tasks, and second, the fact that these tasks are typically performed using latent knowledge in our mind that is impossible to express in words and write using human-readable rules.

✉ Anis Yazidi
anis.yazidi@oslomet.no

Ramesh Upreti
rameshupreti321@gmail.com

Pedro G. Lind
pedro.lind@oslomet.no

Ahmed Elmokashfi
ahmed@simula.no

[1] Department of Computer Science, OsloMet Oslo Metropolitan University, Oslo, Norway

[2] NordSTAR - Nordic Center for Sustainable and Trustworthy AI Research, Oslo, Norway

[3] Simula Metropolitan Center for Digital Engineering, Oslo, Norway

This shortcoming is what motivated the emergence of machine learning (ML) approaches. ML is often referred to as the ability of the machine (computer) to write programs that are data-driven by observing many instances of input and output pairs. A ML algorithm can automatically learn the rules from data, including information from both input and output stages [4]. Its quality depends on the quality and availability of data, and the learning process is guided by loss function or reward values to improve performance in each iteration. In recent few years, ML and deep learning have aroused great interest in different communities. Research community and industries have used ML/AI to solve different types of real-world problems.

ML algorithms are generally categorized into three groups:

As the name suggests, supervised ML is guided by a target variable, also known as the output or response variable. The model is trained based on a subset of the available data, including both the inputs and the corresponding outputs, also called labeled data. Using this subset of the available data, the (supervised) algorithm learns the relationship between inputs and outputs. Supervised ML is generally used for classification and regression problems.

In contrast to supervised learning, unsupervised ML methods do not require labeled outputs. The model is only trained using input data and aims to extract hidden patterns (or relationships) in the input data. Some of the common use cases of unsupervised learning are customer segmentation, frequent item set mining, and anomaly detection [5]. Important methods in unsupervised learning are the so-called classical clustering algorithms, such as *k-means*, density-based spatial clustering of applications with noise (*DBSCAN*) and hierarchical clustering algorithms. Additionally, there are also more advanced approaches, such as deep learning autoencoder algorithms.

Reinforcement learning is a branch of ML which tries to mimic some features of human learning processes: instead of learning from supervised data, it learns from its own experience, composed of a succession of trials, hits, and errors. It works in a feedback-based process, where the agent sequentially performs an action based on the feedback received from previous trials, in the form of a reward or penalty. Consequently, the reinforcement learning agent learns the policy or strategy which maximizes the total reward over time. This type of learning is widely used in AI applications such as self-driving cars and games [5].

Higher or lower levels of trustworthiness of ML approaches are closely related to specific requirements regarding security and privacy issues. Some important surveys have been published recently, covering for instance important approaches within privacy, such as federated learning methods [6]. However, AI trustworthiness, and consequently security and privacy, include other important aspects that should not be considered separately from each other. Indeed, there is an increasing need to establish a more ethical, lawful, and robust framework for the different stages in the AI development life cycle, from design to development to deployment to use. In order to harness the potential of AI in addressing all these aspects, an ideal solution must minimize risk and simultaneously build trust between the different parties that intersect in the use of AI tools. The general term coined to express these recent trends in AI is *Trustworthy AI* (TAI) [7, 8].

The main aim of this article is to provide a survey, addressing not only security and privacy as key aspects of AI trustworthiness but also to discuss their interplay with other aspects of trustworthiness. In particular, we identify (i) a lack of investigations of possible trade-offs between different—eventually competing—aspects of AI trustworthiness in the context of security and privacy and (ii) some important shortcomings in what concerns a unified approach, combining different aspects and their respective trade-offs when assessing the overall trustworthiness of specific AI methods or algorithms. We start with Sects. 2 and 3, where we describe the state-of-the-art approaches to address security and privacy of ML systems, respectively. In Sect. 4, we focus on ML models to other aspects of trustworthiness, presenting a panoply including some of the most used tools to address such aspects. Section 5 outlines the specific topics in each of the previous sections addressing the interplay between security and privacy with the different trustworthiness aspects of AI. Section 6 concludes our survey by discussing some of the literature gaps related to these topics. In particular, we argue that the optimal effectiveness of TAI cannot be achieved without combining AI research in the context of security, privacy, and trustworthiness. Figure 1 sketches the structure of our survey.

2 Security of machine learning

The current trend of ML/AI is more focused on learning from a massive amount of data efficiently, reducing cost, and improving model accuracy and less on designing models while keeping in mind possible security issues.

At the same time, the models for critical decisions are typically vulnerable to attacks [9, 10]. Secure ML has not been as well researched as it needs to be in both academia and industry. In this section, we will explain some of the attacks and potential defense techniques to resolve this issue.

2.1 Categories of attacks

Based on their nature, attacks are organized into six groups according to three categories, namely their influence, their specificity, and their ability to violate security [11, 12, 51]. Table 1 gives an overview of the literature concerning each of these groups.

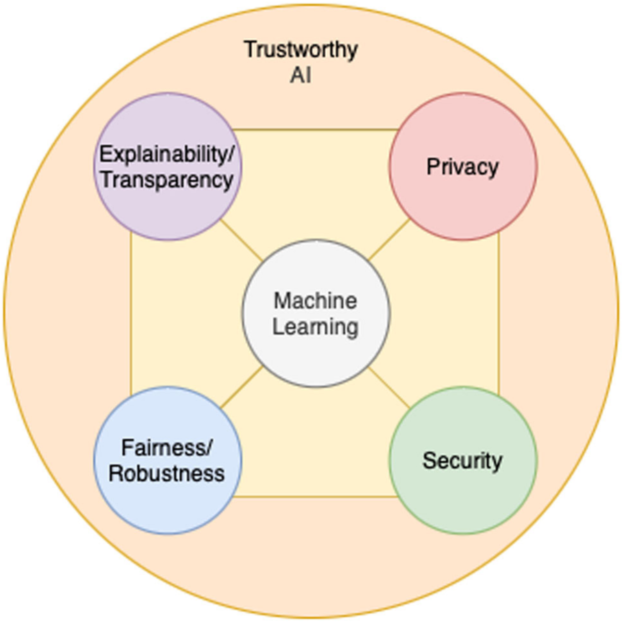

Fig. 1 Scope of the survey: the interplay between security, privacy, explainability, transparency, fairness, robustness and machine learning in the context of trustworthy AI

2.1.1 Influence

The aim of an influence attack is to influence the classifier. The influence can be done in two ways:

- *Causative*: In a causative attack, the attacker has capabilities to modify the distribution of training data. The attacker accesses the training data and manipulates the number of samples in a way that degrades the accuracy of the classifier when retraining the model. This manipulation can be performed by adding more malicious samples or by removing certain data. To carry out this attack, the attacker must have access to the location of the training data. This type of attack is also known as a ”data poisoning attack [12, 18]. Typically, to modify the distribution of training data, a causative attacker uses different kinds of techniques, e.g., dictionary attacks, focused attacks, etc. A dictionary attack is a technique based on dictionary words to attack the model. This technique is used in text classification models and most specifically when the attackers do not know any information about text data [46]. A focused attack is typically focused on one type of text. For example, if attackers want to classify spam emails related to the lottery, the attackers use words related to that email only [46].
- *Exploratory*: In exploratory attacks, the attacker explores the decision boundary of the model. The aim is to gain information about the training and test data sets and to identify the decision boundary model. This can be done by, for example, sending tons of inquiries to the model and obtaining information about the statistical features of the training data [52]. Knowing these features and the decision boundary enables the preparation of malicious input which, after being passed to the model, will result in incorrect classification [12, 53, 54].

2.1.2 Specificity

Depending on the specificity, the attack is further divided into two groups [55]:

- *Targeted*: In a targeted attack, the attacker focuses on one particular case and tries to degrade the performance of the model in that particular case [56]. One example is converting ham information as spam information [57]. The ham (i.e., normal) email should be classified as normal, but the attacker modifies the input in a way that the ham will be classified as spam. The attacker focuses only on the ham class. At a deeper level, the attacker may only focus on a specific type of ham instance.
- *Indiscriminate*: In indiscriminate attacks, the attacker targets all types of instances of a particular class [58]. The attacker’s intention is to degrade the model performance, e.g. classify normal emails as spam.

The specificity dimension of an attack usually groups both these types of attacks into what is called an *adversarial attack* [59]. An adversarial attack is an attack where the aim is to ”fool” the ML model by carefully designing or changing

Table 1 Overview of attacks categories, most relevant defense, and the main corresponding bibliographic references

Attack category		Defense strategy	Bib. sources
Influence	Causative	Data sanitization, Security assessment mechanism	[11–19]
	Exploratory	Algorithm robustness enhancement	[11, 12, 20–27]
Specificity	Targeted	Algorithm robustness enhancement	[11, 12, 28–33]
	Indiscriminate	Algorithm robustness enhancement	[11, 12, 34–36]
Security violation	Integrity	Algorithm robustness enhancement, Privacy-preserving techniques	[11, 12, 37–43]
	Availability	Algorithm robustness enhancement	[11, 12, 44–50]

the inputs in a way that the model accepts malicious inputs as normal information and vice versa [60]. A large number of ML models, including current state-of-the-art deep learning models, are vulnerable to adversarial attacks [61]. One example is the work of Szegedy and co-workers [62], where an image of a panda is classified with a confidence level of only 57.7%, whereas the same image is classified as a gibbon with a confidence level of 99.3%.

In another work [63], the authors argue that current state of the art on deep neural network (DNN) models are vulnerable to adversarial attacks, which could lead to serious consequences, for example, the misclassification of objects caused by adversarial attacks on driverless cars leading to accidents.

2.1.3 Security violation

Based on the nature of security violations or security threats, attacks can be categorized into two further classes:

- *Integrity*: Integrity attacks form a type of attack where the attacker's main intention is to increase the number of false negative cases [11]. In the example of ham *versus* spam classification, an integrity attack consists of classifying as many spam samples as possible as ham.
- *Availability*: In one availability attack, the attacker, instead of increasing the number of false negative cases, increases the false-positive cases [11]. In the case of ham and spam classification, the ham class will be flooded with spam cases. Notice that in the case of binary classification, integrity, and availability are, in practice, equivalent.

2.2 Defense methods

Attack vulnerability in ML systems has become a serious issue. Is it safe to use ML models in security and privacy-related applications? To increase the chances of answering this question positively, it is essential to develop different types of defense techniques. In this section, we will list and describe the most popular and effective defense methods:

2.2.1 Data sanitization

Data sanitization is one of the defense techniques that we can use during the training of a model and is focused on detecting and removing the malicious data in the training set. It is especially effective for causative attacks. Reminiscent of the GIGO concept - "Garbage In, Garbage Out" - usually attributed to IBM programmer George Fuechsel [64], the data sanitization defense method helps to reduce the amount of garbage introduced into the model. One important example of data sanitization method is the so-called "Reject On Negative Impact" (RONI) defense method, introduced by Barreno and co-workers [11] and mainly applied to classification ML models. In the RONI method, one measures the effectiveness of each training instance on the training process and removes the data samples with the highest negative impact on the model from the training set [11, 65]. The training of the model starts with the base model and sequentially adds new instances, checking for a change in accuracy, and adding or rejecting the new instance depending on the model's decision.

Sanitization methods such as RONI are suitable for resolving causative attacks. For example, Barreno et al. applied RONI to a causative attack situation on the *SpamBayes*[1] model [46], showing that while without any defense, the spamBayes model loses considerable accuracy, with the RONI method it is able to avoid this accuracy loss. More precisely, before the defense method, the model showed 97% accuracy classifying ham data and 80% accuracy in spam data, whereas after applying the RONI defense, the model is able to detect 87% of spam and 95% ham. A dictionary attack and a focused attack were applied to the SpamBayes model before and after the RONI defense; interestingly, after the RONI defense method, the model yields better accuracy results.

2.2.2 Disinformation

Disinformation is the process of misguiding the attacker by providing false information or hiding some of the information so that the attacker cannot learn the decision boundary. This type of method is especially helpful against exploratory attacks, where the main intention of the attacker is to learn the decision boundary of the model. The main objective of this method is to confuse the attacker [51, 66].

Disinformation is a simple yet effective defense method, especially in hiding personal information like voter information. One of the interesting works in this area is that of Baloun et al [67]. In this paper, the authors argue that current statistical and deep learning-based models can uncover personal information in the training data. To overcome this issue, the authors proposed what they call a k-*anonymity* technique (also referred to as hiding in the crowd). The main idea of k-anonymity is to hide and provide the model with disinformation, and k refers to the number of (anonymous) groups. This method, instead of providing the real information to the model, converts the data into k anonymous group. E.g. a variable such as a person's age can be converted into ranges $age(0-5, 5-10, .., 95-100)$. The k-anonymity technique has also been used as a privacy-preserving method a lot of research, such as Ref. [68, 69].

[1] http://spambayes.sourceforge.net/

2.2.3 Feature selection

Feature selection helps to extract meaningful features from the data sets. Instead of using all the features, it is more effective to extract important features and use them in the model. For example, in the case of spam or ham classification, we can use different feature selection techniques, such as word frequency count, binary feature representation (if a word exists or not), and the so-called N-*gram* word (sequence of N words) frequency count [11].

The work of Globerson and Roweis [70] is considered to be one of the first and most interesting papers to use this defense method. The authors argue that ML models assign higher weights to some of the important features and lower weights to the other features [71]. This defense method mitigates the ML attacks caused by overloading or detecting the important features in evaluation or test sets. However, it has a drawback: it reduces the robustness of the model and has a greater possibility of being attacked. To overcome this shortcoming, the authors calculate the robustness score using a game theory base model to assign the weights to the features so that no single feature will be overweighted [70].

2.2.4 Randomization

Since the trained model can reveal information about the training data, the randomization method aims at mitigating the ability to learn the correct decision boundary [11].

Randomization is one of the most widely used defensive techniques to mitigate ML attacks, particularly important in cases of overfitting models. Some of the representative works concerning this defense method include [72–77], among others. In a representative work by Pinot and co-workers [77], the authors argue that randomized base classifiers outperform any deterministic model in adversarial attacks. To demonstrate this, the authors propose using randomization techniques based on game theory and show that their proposed defense method achieves an accuracy score of 0.55 in adversarial training, while without randomization, the model achieves a score of only 0.42.

2.2.5 Algorithm robustness enhancement

Increasing the robustness of the algorithm is another possible defense method, with the aim of making the algorithm more accurate in classifying malicious data. Important examples are the bootstrap aggregation method and the random subspace method [11].

One important example of this defense method is the work by Nicolas et al. [63] on adversarial attacks. Here, the authors propose a robustness enhancement method, which they call *defensive distillation technique*. In this technique, the DNN model extracts additional information in the form of probability vectors. These vectors are then fed back into the training process and consequently help improve the generability of the model and reduce the sensitivity to adversarial inputs. The authors report impressive results: the success rate of the adversarial sample reduces from 85.89%, without ”distillation” to 0.45% with distillation.

2.2.6 Security assessment mechanism

Security assessment relies on predictions of the kinds of attacks that may occur. The model designer will first think from that adversarial perspective and then try to propose a solution to tackle or prevent such attacks. There are two solutions to combatting such attacks: proactive defense and reactive defense [12]. Reactive defense is the standard approach in cybersecurity: when an attack is detected, a defense protocol is initiated [78]. One natural drawback of reactive defense is that it neglects possible preventive measures before an attack occurs. To this end, proactive defense strategies have been developed. Proactive defense generally refers to the types of defense techniques that are applied before the system or model is attacked. This defense method is well suited to prevent the attacker from accessing, for example, training data, which are known as *causative attacks*.

An interesting work in this scope is the one by Goodfellow and co-workers [61], as it addresses adversarial attacks. The authors introduce a simple technique which they call ”training with adversarial samples”, where the authors generate data samples by adding linear noise to the original samples. The model is then trained with both original and generated samples, enabling it to also learn the patterns induced by the adversarial attack. Results showed that training with adversarial samples yielded better results, even when the test phase was performed for both the adversarial and normal samples.

3 Privacy of machine learning

Privacy-preserving techniques focus on hiding the real data from the attacker so that it becomes harder for an attacker to predict the data. Some of the most popular privacy techniques are HE, differential privacy, quantization, and hashing [12]. In HE, as its name implies, one encrypts the data using a private key so that no one can access the data. This method is elegant thanks to its ability to perform arithmetic operations on encrypted data [79]. Differential privacy is a simple yet effective method that makes use of noise to ”hide” the model parameters so that it becomes difficult for others to estimate or guess their values [80]. Both HE and differential privacy are explained in more detail in Sec. 3.3. As for quantization, it is a technique that converts the continuous infinite values into small discrete values [81], whereas hashing is a similar concept to encryption, in which one considers transforming larger inputs into small values, and the transformation to hide and return to original values through the respective inverse transformation is performed with the help of a hash table [82].

In the context of privacy-preserving techniques, one recent work by Xu et al. [83] has proposed an interesting illustrative solution. Xu and co-workers emphasize that maintaining the balance between data insights and privacy is an important objective for current ML developers, and thus, to achieve an optimal balance, Xu and co-workers use a differential privacy method to add noise to the hyperplane, defining a support vector machine model. In this way, the authors achieved higher accuracy in classification tasks while preserving user privacy.

The Android operating system is one of the most widely used operating systems in smartphones, wearable devices, IoT devices, etc. To provide different types of features and a better user experience on Android OS, Google needs a lot of user data, however, it has been a big challenge for Google to collect user data due to privacy issues, new laws, and the complexity of storing and processing user data. Moreover, several studies show that more data will result in a better model. Therefore, Google needed an efficient solution to deal with these problems. In 2016, Google's research team H. Brendam McMahan, Eider Moore, Daniel Ramage, Setha Hampson, and Blaise Aguera y Arcas came across a new solution to preserve privacy while leveraging the data from the devices of its users. They coined this new approach Federated Learning (FL) [84].

FL is the new paradigm in the ML family. In FL, the user no longer needs to share the data, as the data are always with the users. FL introduces the concept of sharing the model parameters instead of data. Therefore, it is also called the learning-by-parameters approach. In this approach, the global model is created by the server and shared with all users. Then, each user trains the model with local data on their own device and sends the model train parameters to the server. The server receives the parameters from each user, applies the aggregation to the parameters and updates the model parameters. The updated set of parameters will be shared with all users for the next round. The process will continue until convergence, after some pre-defined number of iterations or in a periodic fashion.

An important feature is that the training process is shifted from the central server to each user device (local device). Initially, FL was introduced for smartphone application by Google, but its applicability is equally important in many different contexts, e.g. in hospitals, in banks, and for the internet of things.

3.1 How federated learning works

In this section, we will explain the training process of the FL system in a step-by-step manner. As shown in Fig. 2, in the first step, a global model is created in the server based on the task, which could be a deep learning model or an ML model like logistic regression. At this point, end devices or data participants have no idea about the model. However, each device contains locally stored data. In the second step, a server sends a copy of the model to all the end users. Based on the nature of the application, the end-user may be located in a different city, country, or region.

Figure 2 illustrates that in the third step, each device has a copy of the global model and local data. So far, no training process has taken place. Now in step 4, each device trains the model on local data and updates the model parameters based on local loss [91, 92].

Once the model updates the gradient/weight locally, the elegance of the model is manifested in the next step. Instead of sharing the data, the model only shares the parameters (weights) with the server. As shown in Fig. 2, in the fifth step, each device sends locally updated weights to the server. If thousands/millions of devices are involved in the training process, then we can imagine that a lot of communication is going to happen between end devices and servers. This is one of the core challenges of FL, which we will explain in more detail later in Sect. 3.4.1.

In the sixth step, the server receives the model parameters from all the devices involved in the training process. Now, the server applies the aggregation on the received parameters, i.e., it calculates the average of all parameters and updates the global model parameters with newly aggregated parameters. All these processes (steps 1–6) happen in one iteration. From the next iteration, the learning process will repeat steps 2 to 6 for a predefined number of iterations or until the model converges.

3.2 Types of federated learning

FL was introduced by Google in 2016; however, this was subsequently further extended by many other researchers. Of the many extended works, [86] is considered as one of the major extensions where authors introduced different types of possible FL systems. Different types of FL systems and where these types of models are more applicable are discussed in the section below.

3.2.1 Horizontal federated learning (HFL)

HFL is introduced in a scenario where two or more participants (data owners) belong to the same field. This type of model is also called sample-based FL because in this FL system data owners have the same feature space, but they have different sample space. Let us explain with an example. A very common example is the next word suggestion on the mobile device. Another popular example is two regional banks that have a different set of samples but have the same

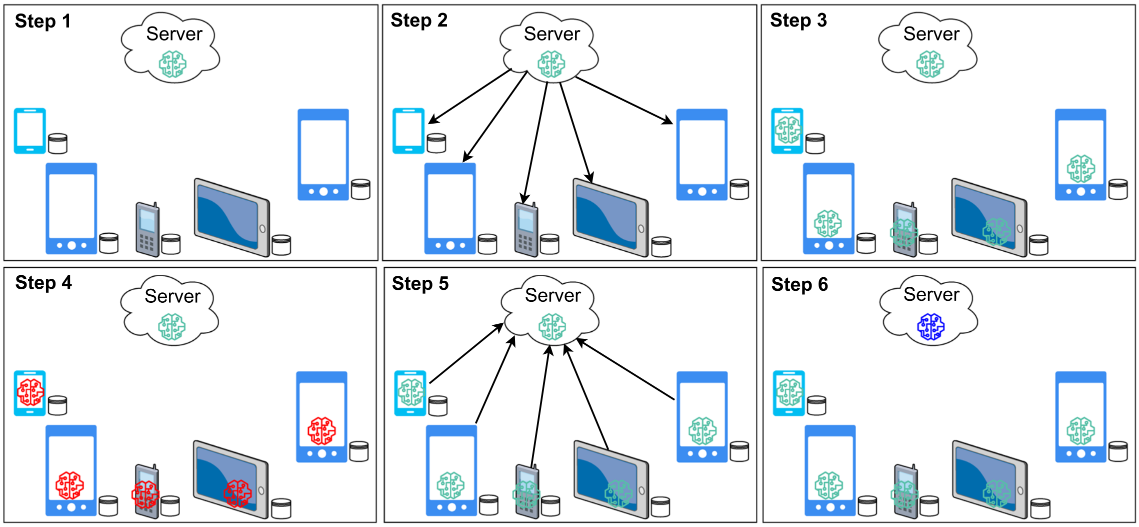

Fig. 2 The different steps in FL training. Step 1: server has initial global model, Step 2: server sends initial model to all connected devices, Step 4: each device receives a copy of global model, Step 4: each device trains the model with local data and updates the model parameter based on local loss, Step 5: each device sends model parameters back to the server, Step 6: server aggregates model parameters received from multiple clients and updates the global model

features [85, 86]. Figure 3 (upper left) is pictorial representation of HFL where we can see that both data participants have different user groups (samples) but their feature space is the same.

3.2.2 Vertical federated learning (VFL)

If one wants to build an advanced model by combining the features from the different data owners, then this type of model is called VFL. Here different data owners mean that they belong to two different service sectors within the same region, for example, a bank and an e-commerce company. As they provide the service in the same region, they have the same user sample. However, their nature of business is different, so they have different feature sets. Therefore, this type of learning is also called feature-based FL. One common use case of this model is an advanced recommendation system for e-commerce sites that combines the user's bank transaction data, which gives more insights into the user regarding the user's purchasing capacity, online shopping habits, etc. [85, 86]. Figure 3 (upper right) illustrates the concept of VFL, in which we can see that they have the same user groups (samples), but their feature space is different.

3.2.3 Federated transfer learning (FTL)

FTL is the extension of vertical FL where the data owner not only differs in the service sector but also belongs to two different regions. Therefore, they have different user groups and different feature sets, for example, a bank located in Norway and an e-commerce site located in the USA. However, because of globalization, they might have a very small group of users in common, as well as a few common features. The federated transfer learning model is built to learn the common representation of features from different data owners based on common user groups, which will help to minimize the prediction of error [85, 86]. Figure 3 (bottom) helps to simplify the concept of FTL, where we can see that a small portion of users (samples) and features are common in data participants.

3.3 Privacy approaches aside federated learning

So far, we have described FL as a strategy to develop an ML model while protecting user privacy. In the following, we shall go into more detail regarding the efficacy of FL. There is no denying that the novel training paradigm offered by FL provides superior privacy to conventional methods, yet FL is not completely privacy-proof. Personal information leaks are still a possibility. The aggregation server or nodes may have vulnerabilities in the FL system. One of these locations may be compromised by malicious attackers, or the node or server itself may behave maliciously by acting as an attacker. In these circumstances, numerous research studies demonstrate that by carefully analyzing the model parameter updates [90, 93], information about a user may be leaked. Through the use of both white-box and black-box access, the attacker has the potential to reveal the data. Therefore, FL itself is not enough for highly sensitive data. There are some advanced techniques that can be used on top of the FL system to further

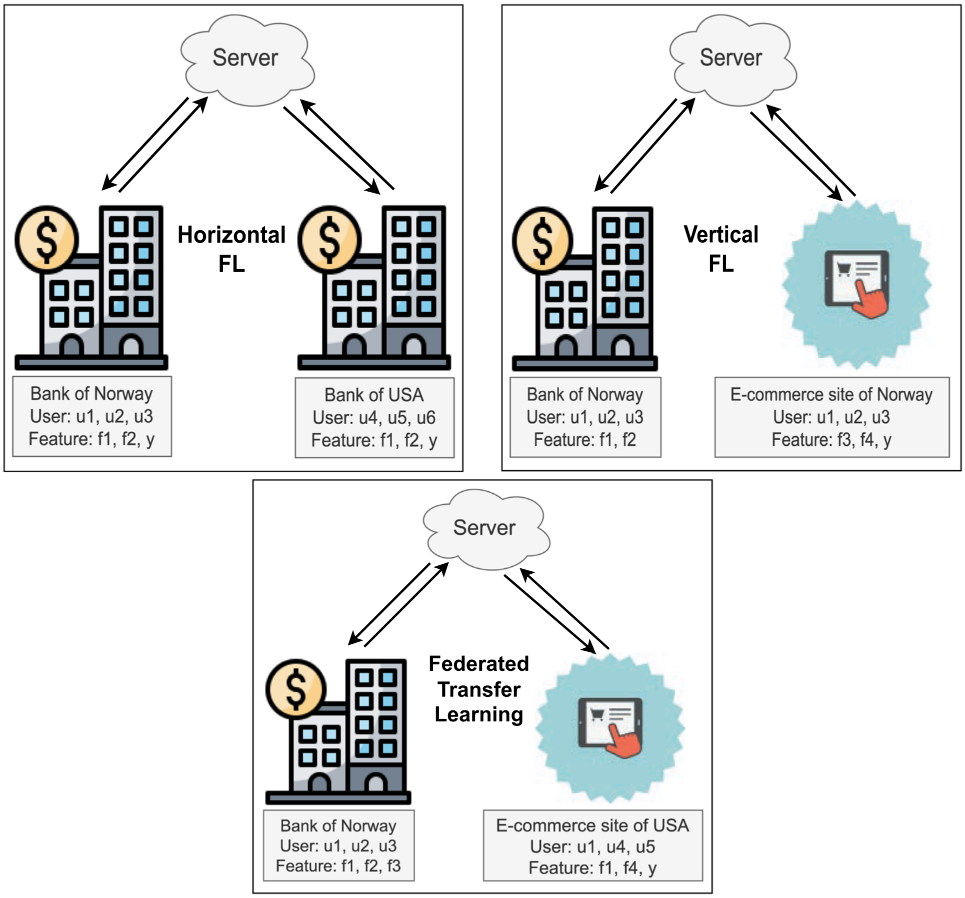

Fig. 3 (Upper left) Horizontal federated learning (HFL). (Upper right) Vertical federated learning (VFL). (Bottom) Federated transfer learning (FTL)

improve the privacy of data. We would like to describe some of the most widely used techniques in this section. Table 2 provides a summary of the content described in this Section.

3.3.1 Secure multiparty computation (SMC)

Suppose you and two of your friends want to calculate the average salary of all three of you. However, nobody wants to disclose their own salary to others. In this case, you can use an external person. Each of you will tell your salary to that person, and they will calculate the average and tell everyone the average salary. In this case, the privacy of your salary depends on the trustworthiness of the external person who may disclose your salary. One efficient solution to this problem is that each one of you split your salary into three random numbers (eg: 50k=10k+15k+25K). You share two numbers with two of your friends and keep one number on hand. The same procedure is followed by your friends. Now, each one of you has one part of your salary plus your friends' part of the salary. Based on this insufficient income information, no one can estimate anyone's salary. Now everyone can sum what they receive with the remaining one part of their own salary. The amount is no longer equal to your actual wage. If everyone sends their salary to the third party to calculate the average, the third party cannot guess anyone's salary, and at the same time you and your friends will know the average salary of the three. This is unbelievable, as without knowing each other's' salaries, you can calculate the average salary. So, you can see how privacy has been maintained. This type of method is called an SMC. The research community has used this method in FL to further enhance privacy [87]. The weight parameters sent to the aggregate server are no longer their own weights. On the other hand, other data owners also receive incomplete information, so no one can guess your model parameters. Can you see any problems with this method? What do you think, is it efficient? Or can we use it in a large network? The answer is no. Due to the communication and computation overhead of each node, this method is not efficient for use in a large network (data participants). However, we can use this method when building the model based on a few data owners. Figure 4 (upper left) illustrates the SMC working mechanism visually.

3.3.2 Homomorphic encryption (HE)

Let's use the same case to calculate the average salary of you and your friends. This time, you and your friends use an encryption technique. Everyone encrypts their salary using

Table 2 Overview of privacy methods, and their main pros and cons with bibliographic sources

Privacy methods		Pros	Cons	Bib. sources
Federated Learning	HFL	• comparatively easy to implement • easily handle varied number of data owner	• malicious aggregator server	[85, 86]
	VFL	• cross organization • extra features	• extra overhead of preparing data • complexity proportional to no. of data owner	[85, 86]
	FTL	• extra information • more generalized system	• highly complex	[85, 86]
Other	SMC	• simple but more effective	• communication overhead • not suitable for larger clients	[87]
	HE	• most secure	• slow process	
	DP	• simple yet effective	• difficult to adjust right amount of noise	[88–90]

the same encryption method. Now everyone sends their encrypted salary to the server. The server does not have a decryption key. Therefore, the server cannot see anyone's salary. However, the server can perform the aggregation operation without decrypting any of the encrypted data. The server sends the encrypted average salary to each of you. You and your friends have their own secret key so that anyone can decrypt it with their secret key and determine the average salary. This type of encryption method is called HE and is widely used in the field of cloud computing. In the traditional encryption method, you need to decrypt the data to perform any operation, yet decrypting data violates your privacy. This is the beauty of HE, which can perform the operation on encrypted data. Despite its very powerful approach, this method is extremely slow and computationally expensive, which is one of the major issues of HE. Figure 4 (upper right) is a visual representation of the HE operating principle, where $sk1$, $sk2$, $sk3$ are the secret keys of individuals used to decrypt messages.

3.3.3 Differential privacy (DP)

Let us continue with the example of calculating the average salary without revealing it to others. As compared to the other two methods discussed in the previous section, this time we will use a very simple method. In this method, everyone adds noise to their salary, e.g., if your salary is 50K, you add 5k as noise. Now your salary is 50k+5K. Everyone adds noise to their salary and sends it to the server. As you have added noise to the data, the data that you have sent is no longer your actual data. So, it is difficult for the server to guess your actual salary. The server calculates the average salary and sends it back to you and your friends. This time the average salary is not the actual average salary, but it is very close to the actual salary. So, you will get an intuitively average salary. This type of method is called differential privacy. DP is one of the most widely used privacy methods in FL [88–90]. DP is very fast and efficient in a large network. If you are worried about the accuracy of the model, then you would be correct; this method always has to trade-off between privacy and accuracy. If you want more privacy, you will lose the accuracy of the model, and vice versa. Further along, in Sect. 3.5 we will discuss the experimental results of different research papers that have used FL and different privacy methods. Here the degree of accuracy possible with these types of methods will be shown. We have tried to summarize the working principle of DP in Fig. 4 (bottom).

In this section, we have explained the most widely used privacy methods in FL. However, there are other possible methods. If you are curious to learn more, we recommend you read anonymization [94], quantization [95], and hashing [96] methods as well.

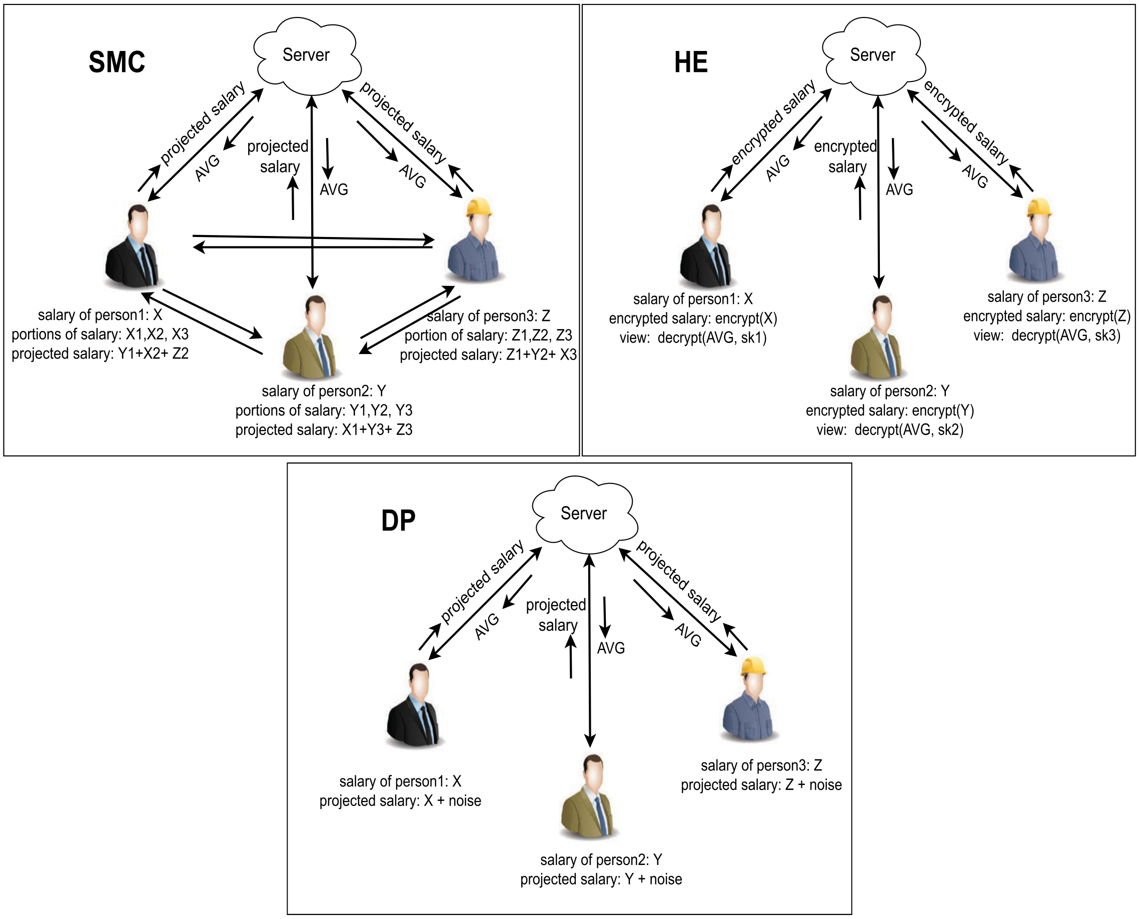

Fig. 4 How can we calculate the average salary of multiple persons without revealing individual salary? We have used the same scenario to demonstrate how each privacy techniques work. Here AVG=$\frac{1}{N}\sum_{i=1}^{N} salary_i$ is a common process in all three methods computed by the server. (Upper left) Secure Multiparty Computation (SMC) working mechanism: each participant splits the salary into a number of participants pieces, keeps one piece self and sends N-1 pieces with N-1 participants, each participant sums the amount and sends the newly summed salary to the server, finally server calculates the aggregate and send back AVG salary to each participant. (Upper right) Homomorphic Encryption (HE) working mechanism: each participant encrypts the salary using HE and sends the encrypted salary to the server, the server performs aggregation on the encrypted salary and sends back to each participant, finally each participant decrypts the encrypted salary using a private secure key (sk_i). (Bottom) DP working mechanism: each participant adds noise to his individual salary and sends noisy salary to the server, the server aggregates noisy salary received from each participant and sends noisy average salary back to each participant

3.4 Core challenges of federated learning and possible solutions

FL is now five years old. In this time, the research community has come up with many interesting ideas to improve the current FL system. However, there are still some important challenges to be solved. In this section, we will explain some of the main challenges of the current FL system that need to be addressed. In addition, in this section we will discuss some of the possible solutions to these problems.

3.4.1 Expensive communication

If you want to build an FL solution between limited data owners (e.g., 3/4 hospitals), then communication is not a problem. However, this is a bottleneck in the case of smartphones, where users are in the millions to billions. If all the users sent updated parameters to the server, it would take a long time to connect with the server, aggregate the parameters, and send them back to each device. In technical terms, this is called high communication latency. To improve the effectiveness of FL, it is very important to come up with a good solution.

One easy and simple solution to this problem is local updating. Instead of sending model parameters at each iteration, we can run the model a certain number of times on a local device and send the parameters to the server. This will help to reduce the total number of communications between devices and servers. Moreover, compression is another possible solution. In this technique, instead of sending all parameters, we can reduce the model parameters or compress them to smaller dimensions and send them to the server. An interesting paper has been published by Google researchers to improve communication efficiency [97]. In this paper, two interesting approaches were proposed. The first approach is the structured update. Within the structured update, low-rank and random mask techniques have been used. In the low-rank techniques, the model parameters are converted in a form of matrix multiplication of two variables ($P = XY$). Here X is a fixed matrix where Y is learned based on the current updates. In simple terms, this method reduces the dimension of parameters so that the size of the updated parameters is small. Similarly, a random mask is another interesting idea. In this technique, the model only sends those parameters that have a weight greater than 0, e.g., if the model has 1000 parameters, but in the current run if 600 parameters have only positive weights and 400 have negative weights, then the model sends only 600 parameters to the server.

The second approach is sketched update. In this approach, the author has used subsampling and probabilistic quantization. In subsampling, the model randomly selects a subset of parameters and sends only those selected parameters, e.g., the model has 1000 parameters, if 550 parameters are selected randomly, then the model only sends the updates of those 550 parameters. On the other hand, probabilistic quantization is a compression technique. Have you noticed one common thing in all these methods? All these methods try to reduce the model's parameter size while sending it to the server. This helps to reduce the uplink communication cost. Maybe you are curious to know if these types of methods help improve the model. If so, we will explain the experimental result of this paper in Sect. 3.5.

Another possible solution to expensive communication is decentralized training. For example, if you want to train the model based on only Norway, we can train the model based on cities, and once the city model is complete, each city will send updates to other cities so that the training of the model is complete on a national level.

Figure 5 shows how the training of devices within four cities can update one global model in a decentralized way. This type of technique is very useful in reducing the number of communications.

3.4.2 System heterogeneity

System heterogeneity is another key challenge of FL, especially in smartphone applications. You may not be familiar with the term system heterogeneity. Let me explain it to you in simple terms: system heterogeneity refers to the differences in system configuration or properties. For example, devices with different network connectivity (3 G, 4 G, 5 G), different memory capacity (RAM:1GB, 2GB, 4GB, 6GB, 8GB, 12GB storage:16GB, 32GB, 64GB, 128GB, 256GB), different CPU capacity, different levels of battery capacity (1600mAh, 2400mAh, 3200mAh, 4000mAh), etc. These types of configurations determine the capacity of smartphones. The smartphone with higher capacity can train, upload, and download the model quickly while it takes a long time for a lower capacity smartphone. Therefore, it is a huge challenge to train the model in varying capacities of devices. To improve the effectiveness of the FL system, it is very important for researchers to come up with a good solution to this problem. We will explain some possible approaches to deal with this problem in the section below.

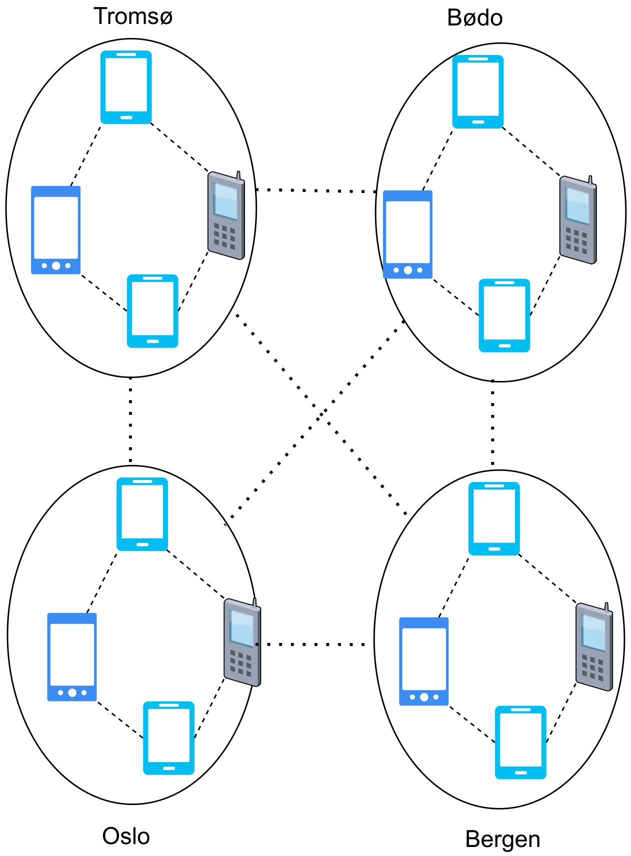

Fig. 5 Decentralized FL model based on four Norwegian cities. First, the user belonging to each city updates the model locally then the global model will update based on each city's local model

One possible solution to this problem is asynchronous communication. In synchronous communication, devices wait for each other, i.e., once the training is completed on all devices, the server will update the model. But in asynchronous communication, the server will not wait for training to be completed on all devices, updates occur in a continuous manner. If 100 devices complete model training, then the server updates the model based on 100 devices and sends it back to those 100 devices. Later, when the other devices complete training, the model will update based on recent server updates. This is an attractive solution to mitigate straggling devices [98]; however, this type of solution has to face a bounded-delay issue [99].

Similarly, active sampling is another possible solution. As the name suggests, the server will select a set of active devices based on system resources. Researchers Nishio and Yonetani have used this method to select only the novel devices for each training iteration [100]. Here, novel devices refer to a system with high resource capacity. Do you think that selecting devices based on only system resources is a good solution? What about the statistical properties of data? You know, some sets of devices may have one statistical property, while others may have another. The research community needs to come up with techniques to perform active sampling based on a combined approach, i.e., considering system resources and statistical properties of data.

Moreover, in the training of each iteration, some devices take a long time to respond, while some may fail due to network or battery issues. Managing this type of situation is called fault tolerance. An easy example of fault tolerance is ignoring these devices during the training process. The devices can be a part of training, but if they fail or do not respond within a predefined time, then the server will update the model based on other devices. Researchers from Stanford University and the University of Southern California (USC) have used the fault tolerance method in FL systems [101].

3.4.3 Statistical heterogeneity

Statistical heterogeneity is one of the key challenges in the application of smartphones. The term statistical heterogeneity refers to the condition of having different data characteristics for a set of user groups. For example, if you want to build the model for the next word suggestion, the text used by people from the USA might be different than that used by people from Asia. Similarly, the nature of text used at night might be different than that used during the day. Moreover, users from different countries use different types of language. Last but not least, the user may use one language to write another language in their way, for example, the expression "*mero naam Ramesh ho*", which is a Nepali way to say the name of one of the authors. Here English text is used, yet the words (mero, naam, ho) do not exist or do not make any sense in English. Building an FL model that can handle the different data characteristics discussed above is a challenging issue.

Researcher Tian Li and his colleague discussed two potential solutions to this problem in their recent paper published in May 2020 [102]. The first potential solution is modeling heterogeneous data. This can be done using techniques called meta-learning and multitask learning. In simple terms, this means building a personalized model based on a global model, i.e., the model is the same, but the parameters have been somewhat optimized based on their personal data. This is also called a device-specific or user-specific model. This solution helps by suggesting the next word, even if the way you write is completely different to others. Another solution is to guarantee the convergence of non-i.i.d data. Here the term non-i.i.d data refers to user's data that are not uniformly distributed in FL, which is one of the key characteristics of FL. To deal with these problems, researchers have used similarity metrics, convexity techniques, etc. Although different solutions have been tested by researchers, the available solutions are not fully reliable and robust.

3.4.4 Privacy concerns

FL has emerged as a solution to privacy issues and leveraging the power of training the model on distributed end devices. In general, there is no doubt that FL provides a high level of privacy. However, if you want to use FL for highly sensitive data among limited users, and if the participant user or server is malicious, it is possible to infer properties of your training data by carefully observing model updates. We have paid special attention to this topic and have dedicated a specific section to it.

3.4.5 Traceability and accountability

When we are building an FL model on a critical dataset, it is very important to make sure that the model improves over time. We need to keep in mind that some malicious data owners may be involved in the training process. The intention of malicious data owners is always to misguide the training process. Therefore, it is very important to keep the track of the contributions of each data owner in the final model, or in other words, of who is more responsible for the unexpected result of the model. Currently, there is no system for the traceability and accountability of the FL model. The research community needs to come up with a feature that is capable of maintaining the accountability of the model.

So far there is no solution to this problem. However, a group of researchers have discussed some potential ideas in [103]. One possible solution could be to build explainability in the model. Another possible solution is to build a tool to keep track of the results, as well as each model contribution. This helps to identify which model is more responsible for false-positive or false-negative cases. Building the tool to ensure data quality before the training process is another interesting solution; however, how it can be done in an efficient way is a topic for future research.

3.5 Discussion of existing privacy preserving solutions

One of the most interesting papers published by Google researchers is [104] where the researcher used an FL model for next word prediction on smartphones [104]. Google was facing two main issues, one was how to preserve user privacy and the other was how to reduce computational cost. In this paper, the authors have used an FL model that can solve those two issues and provide the same level of service or better service. To verify the performance of the model, they have also built the server-based model (traditional approach). The authors have used the experimental approach in this paper. For the experiment, they used two types of datasets (mobile app type logs and mobile app type cache). For the prediction, they built two Coupled Input and Forget Gate (CIFG) models: one for server-based and one for the federated base. CIFG is a recurrent neural network model. Before this model, Google used the N-gram model for this type of model. Therefore, for comparison purposes, the authors also created an N-gram model. Researchers used the federated averaging algorithm in a federated server. The outcome of the experimental result was very interesting. First of all, in all cases, the CIFG model performs better as compared to the N-gram model. Similarly, if we compare the performance of the FL-based model and the server-based model, the author shows that in two of the experiments, the FL-based model outperforms the server-based model, while two other times the performance of the two models was almost the same. This result shows that Google was able to achieve a better model using FL and, at the same time, solve the privacy problem and reduce the computational cost.

Similarly, in [105], researchers try to resolve a real-world problem. Three universities have medical images (MRI diagnosis images) related to prostate cancer segmentation, and they want to build an efficient model for the detection of cancer. However, they had two main issues. The first was in regard to the local data; each university was not able to build an efficient model. They believed they could improve the model by adding more data. However, the second issue raised here was that they could not share the data with other institutions because of patient privacy. To solve this problem, the researchers used the FL model and a modeling and experimental approach. First of all, they built a deep learning model and trained the model on each institution separately based on their local data. They then built an FL model that trains a global model on each institution in parallel and sends the updated parameters to a cloud-based server for aggression. After completing the training process, they obtained an impressive result. To verify the effectiveness of the FL model, they tested their FL model with the ProstateX challenge dataset. Interestingly, the FL model performed better on this dataset as well. This experimental result shows that FL not only helps to preserve privacy but also helps to build a better model with access to more data.

Moreover, [89] is one of the most interesting pieces of research that focuses on improving the privacy of the FL system. The title of the paper is *A Hybrid Approach to Privacy-Preserving Federated Learning* [89], and researchers from IBM and professors from the Georgia Institute of Technology worked together on this paper. In this paper, the authors have pointed out three main motivating factors. First, training the data locally does not guarantee a sufficient level of privacy. Second, SMC is a privacy method but it is vulnerable to inference, and third, DP is another privacy method, but it leads to low accuracy, especially when the number of participants with low data is high. To address these issues, researchers have proposed a hybrid approach. Instead of using SMC or DP only, the researchers combined SMC and DP methods in a way that enhances privacy, and at the same time yields higher accuracy. For different epsilon values (high epsilon = low noise and vice versa), the proposed approach obtained a higher f1-score as compared to the local DP method. As the number of participants increased, the local DP performance dropped significantly, but interestingly the proposed approach yielded the same level of performance with any number of participants. The experiment results show how the inclusion of trust parameters in the proposed approach help to obtain better performance. The researchers tested their proposed approach with a decision tree, a support vector machine, and a convolution neural network.

In [90], the author pointed out that the DP method needs to sacrifice a lot of accuracy to maintain a high level of privacy. Noise is added to each weight equally in DP, and the authors argued that this is the main drawback of DP. Therefore, they proposed their own algorithm called APFL (Adaptive Privacy-preserving Federated Learning), which adds noise based on the contribution of each weight. To calculate the contribution of each weight, they used a layer-wise relevance propagation algorithm. This technique helped to significantly reduce the noise and improved the accuracy of the model. The experimental results show that the proposed APFL model yields higher accuracy compared to existing DP-based research.

4 Machine Learning in the context of Trustworthy AI

In the previous two sections, we presented different types of security and privacy issues raised by ML systems and discussed different types of potential solutions to deal with those problems. We mainly focused on how to improve security and privacy issues, while completely disregarding another aspect: the trustworthiness of the solutions. As security and privacy are the biggest issues in today's ML applications, it is very important to make sure that the decisions made by ML models are trusted. The ML engineer who is involved in the development of a model may have better ideas about how the model makes a decision, but this might be completely unknown for people with a non-technical background like users, stakeholders (owners), lawyers, etc. While working on a critical application, ML-based solutions should maintain trust among different parties. In addition, the new rules and laws (GDPR) also require that if any decision is made based on the ML model, an explanation must be provided. Therefore, this section aims to identify the current research related to trustworthiness in the context of security and privacy.

Trustworthiness, in general, refers to being trusted, reliable, and confident [106]. In the context of ML/AI, trustworthiness not only refers to having an accurate model but also deals with explainability [107], transparency, fairness, winning trust, and robustness [108]. Trustworthiness is often alluded to as Explainable AI, XAI, Responsible AI, etc., which are widely used in the current research domain of AI. In fact, trustworthiness is a broader term than explainable AI. Explainable AI uses natural language and different kinds of visualization tools to explain the rationale based on the context in which the ML model has made the decision [109]. However, Trustworthiness is not only limited to explainability. It deals with winning the trust and confidence of different parties, making models and data preparation transparent, and building a robust and fair model [110]. Therefore, trustworthiness does not only come from building an explainable tool, but it involves several other steps by which ML applications build trust and earn it. The concept of Trustworthy AI is important in several specific contexts, such as health [111].

The meaning of trustworthiness may differ from person to person. For example, for developers, trustworthiness means knowing the quality of data, how the data is prepared, knowing the model architecture, identifying the importance of each feature in relation to output, etc. Similarly, for a user or client trustworthiness means knowing how the model makes the decision and why the system is safe to use. On the other hand, the meaning of trustworthiness for a lawyer is to know the legal justification of the decision and to ensure the rights of users to be informed of any explanations [112].

4.1 Component of trustworthy AI

Trustworthiness is in fact a sort of ”umbrella word” that incorporates different aspects, including Explainable AI and Responsible AI, among others. In the following, we address the central aspects of the current AI literature that constitute the overall concept of Trustworthiness in AI. In particular, we focus on the EU guidelines [7], which refer to the following aspects detailed below.

4.1.1 Lawful AI

Every system that runs in this world falls under certain rules and laws, and an AI system is no exception. There are already different kinds of rules and laws created by the European Union and other responsible agencies at national and international levels. Companies should have to consider these laws in order to develop, deploy, and use AI systems. Developing the AI solution under defined laws helps to maintain trustworthy AI [7].

4.1.2 Ethical AI

AI laws and rules define what one can and cannot do while developing and deploying AI solutions. However, not everything can be covered by laws. In such cases, it is always necessary to think one step ahead of AI laws, such as, for example, the ethical perspective. This is very important while working in critical applications like healthcare. The trustworthiness of AI cannot be achieved without ensuring ethical norms.

The term ethical AI is a broader term; therefore, the following components of trustworthy AI that are commonly discussed in the literature can be categorized under the following ethical AI components:

- *Respect for human autonomy*: The universal truth is that we build AI for human beings. Therefore, AI-based applications should respect the human beings involved in the different stages of development, for example, from someone who participates in data collection to an expert who uses it for making decisions. AI systems should always be designed from human-centric design principles to complement and empower human cognitive, social, and cultural skills [7]. Therefore, it should not replace, manipulate, or herd humans.
- *Safe and secure*: When it comes to the use of AI-based systems, it must be ensured that the system is safe and secure for human beings, that is, it should not cause or exacerbate harm to human beings [7]. A critical example is self-driving cars or the use of robots in working environments; AI systems must not harm the driver or anyone in the surrounding environment. Another important yet critical aspect is that it should not be open to any kind of malicious use.
- *Privacy*: Privacy is an additional essential aspect of trustworthy AI [7, 8]. In today’s digitalized world, user privacy has become a major issue. Different responsible organizations have introduced stronger rules to preserve user privacy. Therefore, AI-based solutions should make sure that the privacy of users is preserved. Attackers use advanced techniques such as adversarial attacks, membership inference attacks, and data linkage to reveal the information of users from AI solutions.
- *Fairness*: The EU guidelines for trustworthy AI mention that AI-based systems must be fair in their different stages, from development and deployment to use [7, 102]. Normally, the term fairness has a substantive and proce-

dural dimension. The term substantive dimension refers to the fact that AI solutions should not be biased, discriminative, or stigmatized toward an individual or a group of people. For example, an AI-based algorithm called COMPAS used to make a judgment decision in a US court was found to be racially biased toward African American defendants [113]. Similarly, fairness is also about the freedom of choice offered to users to decide whether or not their data participates in AI-based systems. Even if the user has granted permission to use the data, there should be a provision to revoke the decision at a later date. On the other hand, the procedural dimension deals with the fairness of the process. Here the process refers to the steps involved before the decision is reached.

- *Accountability*: As we have said before, ethical IA is a very broad topic; accountability also falls under ethical IA [7, 114]. The term "accountability" refers to keeping records of everything from development to deployment so that anyone who asks can audit the documents. Accountability is only possible if the person is honest. For example, if people report only the good side of the algorithm and do not mention the bad side or negative impact, it will be problematic.
- *Explainability*: Explainability is one of the main components of trustworthiness. In the last couple of years, the topic has attracted a great deal of attention from the research community and some promising work has been published. Researchers have mainly focused on building tools to explain the reason behind the model's decisions. Researchers have used different kinds of techniques like gaming to identify the contribution of each feature in the decision and provide explanations using visualization and natural language. This helps to build user confidence. On the other hand, it also helps to diagnose the model and improve it [112].
- *Transparency*: Explaining the model's own decisions is not enough to achieve a high level of trustworthiness. In the case of trustworthiness, it is very important to gain the trust of the different parties. This can be achieved by making transparent how we choose the model, what the architecture of the model is, what parameters are used in the model, what the main function of each layer is, how the data is prepared, what the statistics of the training data are, how the features are selected, etc. This kind of information provides more insights into data and the decision process [112].

4.1.3 Robust AI

Even if the AI solution manages to pass laws and ethical standards in this regard, it is equally important to have a robust AI model. This becomes mandatory when AI is used in critical applications like self-driving cars, autonomous weapons, etc. The robustness of AI applications ensures that they will not harm anyone unintentionally. In other words, robust AI refers to an AI model that is mathematically verified or has been validated with different kinds of tests, and ensures safety. The robustness of AI solutions can be verified from both a technical and social perspective [7].

In the literature, researchers also often use the term generalizability as a component of a trustworthy AI, related to the robustness of the tool or method. Generalizability is an important aspect of AI mostly due to the fact that models learn from data and make decisions. Therefore, the availability of sufficient data leads to a better model, i.e., the more data we have, the more generalized the model we obtain. However, data deficiency is one of the biggest issues in the current AI field. Therefore, it is very important to be transparent about how many records are used in training, and the number of records belonging to each class (if it is a classification model). In the case of low data availability, it is also important to divulge what kind of techniques are used to increase the number of records. Providing all this information to the public helps gain the trust of users [112].

4.2 Tools to assess trustworthiness of AI methods

As the field of trustworthy AI is in an evolving phase, it is very challenging to measure whether each component of trustworthy AI is taken care of or not during the development of AI solutions. In fact, there are not enough reliable resources available in the market. Building tools for measuring/assessing trustworthy AI components is an active field of research. In this section, we explain some of the tools and assessment methods available on the market.

4.2.1 Assessment list of trustworthy AI

After the European Union (EU) introduced ethical guidelines for trustworthy AI, they then also introduced the Assessment List for Trustworthy AI (ALTAI) to make the development of AI solutions more responsible and sustainable in Europe [115]. The EU believes that the introduced assessment list helps organizations to build AI by keeping ethics as a central pillar of development, which will ultimately benefit individuals and society. The ALTAI was introduced by experts from multidisciplinary teams such as AI designers, AI developers, data scientists, legal officers, management, and so on, and the assessment list is available both offline and online. The assessment list is not a mathematical tool that provides a score to the developer to measure each component of TAI; indeed, it is a list of questions or guidelines which guide the AI developer at different stages of development by asking questions related to the requirement of trustworthy AI. The EU ALTAI is based on seven different requirements, as follows:

- Human agency and oversight: questions related to human agency, human autonomy, and human oversight.
- Technical robustness and safety: questions related to resilience to attack and security, general safety, accuracy, reliability, Fall-back plans, and reproducibility.
- Privacy and data governance: questions related to privacy and data governance.
- Transparency: questions related to traceability, explainability, and communication.
- Diversity, non-discrimination, and fairness: questions related to avoidance of unfair bias, accessibility and universal design, and stakeholder participation.
- Societal and environmental well-being: questions related to environmental well-being, impact on work and skills, impact on society at larger or democracy.
- Accountability: questions related to auditability and risk management.

4.2.2 Adversarial robustness toolbox (ART)

This tool was developed by IBM researchers, and it helps to measure the security of ML applications [116, 117]. It is available as a Python library and helps AI developers and researchers defend and evaluate their ML models and applications against different kinds of attacks, as we discussed in the previous section on ML attacks. This tool can be used in all kinds of ML frameworks available in the market (scikit-learn, PyTorch, TemsorFlow, etc.). Similarly, it supports all kinds of data (tables, images, video, audio, etc.) and can be used in different ML tasks (classification, regression, generation, etc.) [116].

4.2.3 AI privacy 360

AI Privacy 360 helps AI developers implement any relevant privacy requirements. As we know, privacy is always compromised with accuracy. The AI Privacy 360 tool helps maintain a suitable balance between privacy, accuracy, and performance at the different stages of development [118]. This tool is available as a Python package and supports all available ML frameworks for different kinds of data and tasks.

4.2.4 AI fairness 360

This is another great tool designed and built by IBM researchers. This is a Python-based open-source toolkit that helps to verify and mitigate the unwanted bias of datasets, ML models, and state-of-the-art algorithms. It provides a wide variety of fairness metrics (70 metrics, in fact) and 10 different bias mitigation algorithms. This tool helps AI developers design and build fair AI applications [119].

4.2.5 AI factSheets 360

AI FactSheets 360 is a great tool for evaluating the transparency components of trustworthy AI [120, 121]. As the name suggests, this toolkit generates a factsheet by outlining the details of the dataset used to train the model, what the data selection criteria, how the model was trained and tested, model robustness, fairness test, robustness test, privacy test, safety and security test, different kinds of performance metrics, etc. All details related to AI applications can be included in the factsheet so that anyone can easily access it to understand the work behind it.

4.2.6 Uncertainty quantification 360

Uncertainty Quantification 360 is a very useful tool for assessing and mitigating uncertainty in AI models [122]. It is an open-source Python-based library that provides flexibility for AI developers to measure uncertainty using a diverse set of algorithms. This tool also offers the possibility to improve uncertainty quantification during the development of AI applications.

4.2.7 Explainability

Of the different components of trustworthy AI, explainability is the one that receives the most attention from researchers, and there is a lot of research in the field of explainability assessment. Therefore, we would like to explain some of the most commonly used tools for explainability assessment of AI models, namely:

- *SHAP*: SHAP (SHapley Additive exPlanations) is one of the most widely used tools in explaining model decisions. SHAP is based on a game theory concept called shapely value. SHAP basically calculates the contribution of each feature in a collaborative way [123]. SHAP uses different kinds of visualization tools to explain the model output. The benefit of using this tool is that it supports any kind of ML and deep learning models for tabular, image, and text data.
- *LIME*: LIME (Local Interpretable Model-Agnostic) follows a similar concept to SHAP. SHAP uses a concept of game theory and calculates the shapely values, while LIME perturbs the inputs around its neighbors and calculates the contribution of each feature. Therefore, it is also called a surrogate model, i.e., it uses the black-box ML model to calculate feature contribution. LIME is widely used for classification and regression problems. The extension of LIME is called SP-LIME, which selects a sample explanation from a set of explanations [124].
- *AI Explainability 360*: AI Explainability 360 is an open-source tool kit developed by IBM (also called IBM 360)

to support the interpretability and explainability of any state-of-the-art ML models. This tool is more enriched than the previous two tools because it supports ten different state-of-the-art explainability algorithms, including SHAP and LIME [125, 126].Therefore, users get the flexibility to choose different algorithms based on their requirements. Another crucial benefit of using this tool is that it provides directly interpretable local post-hoc, global post-hoc, self-explanations and metrics for the data. As compared to other tools available in the market, this is the only tool that provides those features. This tool can be used by data scientists, decision-makers, regulators, and users.

4.3 Recent works in trustworthy AI

The number of publications related to reliable Artificial Intelligence has increased significantly, as the topic has received a great deal of attention from the research community, industry, and governments. Therefore, in this section, we would like to mention some of the recent papers representing this current trend. Recently, Li Bo and his colleagues published a review article in which they examine the topic of trustworthy AI from both a theoretical and a practical standpoint [114]. The authors discussed the seven principles of trustworthy AI: robustness, generalization, explainability and transparency, reproducibility, fairness, privacy protection, and accountability. In addition, the authors investigated the various techniques utilized by AI application developers at various stages of the AI lifecycle. After reviewing a number of articles, the authors concluded that trustworthiness approaches are still immature and lack standardization. Similarly, the authors argued that the methods used to assess credibility are insufficient. This review paper provides an overview of current research in trustworthy AI; however, the authors have not investigated the trade-offs between different trustworthy AI components.

The work of Haochen et al. [8] is another recent paper on reliable AI. The authors agree with the enormous benefits provided by artificial intelligence technology and, at the same time, are aware of the recent unintended harms to humans caused by AI. Therefore, the authors argued that transparency in AI should become a compulsory part of AI development. According to the author, TAI is a very large and complex subject. Therefore, in order to make the TAI understandable, the authors have introduced the six most crucial dimensions of TAI. They are safety and robustness, nondiscrimination and fairness, explainability, privacy, accountability and audibility, and environmental well-being [8]. Another interesting paper by Kush [127] emphasized that trustworthy ML and AI goes far beyond making highly accurate models. To make the AI trustworthy, developers need to consider data shift robustness, protection from data poisoning, fairness, interpretability, system level transparency, end-to-end service level provenance, etc.

Luca Vigano and Daniele came up with the idea of explainable security [128]. The authors were inspired by the DARPA's XAI [129] and proposed a new paradigm called Explainable Security (XSec) in the field of security. Instead of only focusing on the explanation of how the model makes a decision, the authors presented ”Six Ws” as the main component of XSec. The **Who**: designer/developer, users/clients, attacker, analyst, and defender, who are responsible for both providing and receiving the explanations, The **What**: clients' requirements, how to use the system securely, security properties, the system model, the threat model, possible vulnerabilities or attacks, etc., all need to be explained, The **Where**: where the explanation is, is it provided as part of a security/privacy policy, or as an explanation-as-a-service, or it is detached from the system, the **When**: whether an explanation needs to be given during the design, implementation, modification, installation, use, defense, attack, or analysis phase, the **How**: if it could be expressed in the form of natural language, or graphical language, or formal language, or through gamification. The last W is **Why**, where the authors explain why we need XSec. The authors argue that their proposed paradigm provides security by maintaining trust, transparency, confidence, accountability, verifiability, and testability.

Neel et al. [130] have proposed the idea of combining explanations while maintaining privacy. The authors fully agree with the current requirements for algorithmic transparency when using it in critical decision-making domains. However, the authors argue that the model explanation may leak information from the training data [130]. To address this issue, they have proposed the architecture of combining explanation and privacy. The authors have used DP to maintain privacy. For this, they have proposed their own algorithm, called the Adaptive Differentially Private Gradient algorithm. The algorithm adaptively reuses the past DP explanation that helps to reduce the overall privacy loss. This is a nice way of making an explainable model by preserving privacy. DP has always been a trade-off between privacy and accuracy; therefore, it could be interesting to see a trade-off between privacy and explanation addressed in future work.

Similarly, Danilo et al. presented the idea of combining privacy, fairness, and explainability to build a trustworthy learning model [131]. The authors noted that the right to preserve privacy, build a less discriminatory model with sensitive attributes (e.g., facial color, sex), and provide an explanation of the model decision process to users in a single model is the ultimate requirement to ensure trustworthiness. In the proposed architecture, they used HE to encrypt the training data and associated labels, and then use HE in the training and forward model. To ensure fairness, the authors used a Tikhonov regularizer before the final layers. To make the model interpretable, the authors used the Grad-CAM method, which extracts the attention map of the given input image, i.e., it highlights the most influential features of images that can be used in supervised learning. The authors used the output of Grad-CAM for local and global model explanation. To demonstrate the effectiveness of the proposed approach, the authors used a VGG 16 model architecture for the application of face recognition. The authors have shown that it is possible to build a model that can maintain privacy, fairness, and explanation in a single model [131]. However, the authors did not discuss the direction of ML adversarial attack risk as they had already explained the attention mask of model decision.

5 Discussion: assessing the combined interplay of different aspects of AI trustworthiness

In the preceding sections, we covered different components of trustworthy AI based on EU guidelines and state-of-the-art work. Similarly, we also discussed the importance of trustworthy AI evaluation frameworks and presented some of the available frameworks used for evaluating different components. However, we notice that there is a gap in the literature as very few studies have analyzed the interplay between different components of trustworthy AI while building the solution for archiving specific components of trustworthy AI. This interplay has not gained sufficient attention in the literature. Therefore, to draw the attention of researchers and notifying bodies, in this section, we are presenting some of the interplay between different components of trustworthy AI, how a single component of trustworthiness might negatively affect other components, the reason behind it, and the measures that need to be taken to achieve some trade-offs between the different components.

5.1 Possible interplay between trustworthy AI components

As we mentioned earlier, the current research lacks sufficient amount of discussion in this direction. However, we aim to demonstrate the potential interplay, considering the different pieces of available literature and our own understanding.

5.1.1 Privacy vs rest

Privacy has become one of the major concerns in the domain of artificial intelligence-based systems. To make the AI application trustworthy, the privacy measures have to be taken seriously. In Sect. 3 section, we presented different methods used for preserving privacy. However, the consequences of privacy methods have not been sufficiently analyzed.

One of the major trade-offs of privacy methods has to do with the performance of the system. The performance of a system generally refers to the productivity, accuracy, or efficiency of the AI application. Several studies have reported that the performance declines because of privacy methods [132, 133]. The accuracy of the model is mostly affected by DP privacy methods, as the nature of DP is adding noise at different stages.

Another possible interplay of privacy methods is with the accountability/ethical component. For instance, the technical person working on implementing privacy solutions may not have a sufficient understanding of ethical aspects. This could be a huge issue when it comes to critical applications such as healthcare decisions, self-driving cars, or judicial decisions. The potential decrease in performance resulting from the implementation of ML privacy techniques may lead to the adoption of a decision that has inherent risks. This has raised the blame game of who (privacy method developer or model developer) is responsible behind this result [134]. This brings up the moral dilemma of how much privacy is sufficient to achieve the required standard of performance.

Similarly, another major trade-off is with explainability. As we discussed in the previous section, differential privacy is the most widely used method for preserving privacy; however, adding noise to the data or model weight converts the data into noisy form. This raises a huge issue for the explainability model. The explainability model built based on data face the issue of poor explanation. In [135], the authors test the effect of three different privacy methods over explainability models. The experimental result shows that DP hampers the interpretability of explanations. Additionally, the authors reported that the fidelity of explanations is potentially deteriorated when using DP [135]. Similar results are also reported in [136, 137]. In addition to this, we believe inconsistency in explanation will be another major issue because of the non-static noise added each time. All of this raises the question of trustability in the AI explanation system, which can indirectly relate to ethical components. Who will be responsible if privacy techniques result in an incorrect or inconsistent explanation? Is it the team that works on explanation solutions or the team that works on privacy methods?

Privacy component interplay is also related to the fairness component. As the model accuracy drops because of privacy methods, the drop in accuracy will be higher in imbalanced classes. As discussed in [138], the accuracy of minority sample classes further declines because of privacy methods.

The research article [139] published by Google researchers has discussed that the theoretical relationship between privacy and robustness is unknown; however, experimental results have demonstrated that they are mutually detrimental. To demonstrate the effect of DP on robustness, the authors from [140] build the DP model and a non-DP-based model. Five distinct evaluation methods show that the robustness of DP-based models is comparatively lower. The conclusion from [140] is further verified by authors [141], which demonstrates that DP models do not exhibit greater robustness compared to conventionally trained models.

Another major trade-off exists between privacy components and the security of ML systems. Some of the recent works [140, 142–145] demonstrate that privacy methods, especially DP, are more susceptible to poisoning attacks. The poisoning attack is becoming the biggest thread in the ML system for sensitive data under the decentralized training system using local differential privacy. Similarly, [141] reported that an unfavorable choice of parameters in DP training can lead to gradient masking, which can ultimately lead to security risk.

In summary, the previous section has discussed the significant connection between the privacy component and other factors such as performance, explainability, accountability, fairness, robustness, and security. However, it should be noted that the privacy component also indirectly interacts with the remaining components.

5.1.2 Security vs rest

As we discussed in Sect. 2 the security of ML solutions have become one of the biggest threats in recent years. As technology advances, the attacker has become even smarter than before. Even a modest amount of information about ML models or data is sufficient for an attacker to create a targeted attack to leak sensitive data. The ML security is an active field of research. The possible ML attacks and defense discussed in Sect. 2 are just a primary list. Every time new and advanced types of ML models are emerging. As a result, research scientists are more concentrating on building stronger and more efficient security methods. In this context, the ML security interplay with other components of trustworthy AI has not gained enough attention. Nevertheless, a number of studies have indicated that current ML security approaches are prone to hindering performance [146].

Although there is a lack of sufficient exploration, it is highly probable that certain ML defense methods pose challenges for explainability components. For instance, techniques such as denoising model weights or model weight pruning are employed to counter model steal attacks. In the previous study [147], the ability of the DP method to reduce model inversion attacks has been explored; however, the authors did not investigate its impact on the explainability component. Similarly, the DP is one of the most widely used defense approaches against membership attacks [53]. We have already discussed the interplay of DP-based methods on performance and explainability components in the previous section. In the survey paper [53], the authors have discussed model quantization, half-precision floating point, setting all gradients below certain thresholds to zero, etc. as methods for attack against reconstruction loss. However, it is well acknowledged that these methods are associated with trade-offs in terms of both explainability and performance.

Existing algorithms that enhance model robustness against attacks during deployment often come at the expense of compromising data privacy [148]. Previous research has often treated the security and privacy domains separately. Adversarial defense methods aimed at enhancing robustness can paradoxically increase the susceptibility of the target model to membership inference attacks. It was shown in [148] that employing adversarial defenses to train robust models can amplify up to 4.5 times the advantages of membership inference compared to naturally undefended models.

The trade-offs between ML security methods with fairness and robustness have not been explored directly. To the best of our current understanding, there has been a lack of research conducted in this particular area. Nevertheless, it might be postulated that there are trade-offs inherent in their relationship. For instance, in the previous paragraphs, we mentioned that DP is used against model inversion attacks and membership inference attacks. On the other hand, we have already discussed how DP reduces fairness and robustness in the subsection of trade-offs between privacy vs rest. Therefore, in an indirect way, we can link the trade-offs between ML security methods and fairness and robustness.

5.1.3 Explainability/transparency vs rest

The major interplay of explainability and transparency components of trustworthy AI is with privacy component. The research reported in [149] reveals that constraints imposed by trustworthy machine learning on the training process can introduce significant privacy concerns. Achieving trustworthy machine learning necessitates additional model constraints. Specialized algorithms are employed to enable privacy-preservation, fairness, robustness, and explainability. However, these algorithms come with inherent trade-offs. Some recent research has explored the trade-off between trustworthy machine learning and model performance, particularly prediction accuracy. For instance, in [150], the authors demonstrate that the trustworthiness of results is influenced by data quality. In other terms, privacy concerns indirectly impact trustworthiness as a contributing factor. There are also some trade-offs between explainability and privacy. Indeed, according to [149], model explanations can be exploited by inference attacks. From a privacy standpoint, model explanations furnish attackers with supplementary information, especially in scenarios where direct access to a model's uncertainty or loss is limited. High feature attribution values serve as a proxy for model uncertainty, indicating that a small input change could substantially alter the model's output. Consequently, attackers can construct successful membership inference attacks solely by leveraging model explanations to differentiate between members and nonmembers. The issue of information leakage from the explanation dataset arises when explaining model behavior [136]. Conversely, safeguarding sensitive data using certified differential privacy comes at the expense of explanation quality. The inherent randomness of differential privacy algorithms might compromise explanation fidelity due to increased uncertainty in model predictions and local approximations. Some feature-based model explanations, dependent on model parameters [151], [152], can exacerbate privacy vulnerabilities in the context of white-box inference attacks. The study in [153] explores connections between model explanations and the leakage of sensitive training set information. Privacy risks associated with feature-based model explanations are analyzed using membership inference attacks, quantifying how much information about a data-point's presence in the training set leaks through model predictions and explanations. The research underscores that offering model explanations might compromise user privacy. Current model explanation technologies lack provable privacy guarantees. Counterfactual explanations, despite highlighting key features used by black-box models and providing actionable insights, also inadvertently leak information about the model itself, raising privacy concerns [154]. In [155], the authors express concern over the possibility of ”fairwashing” through the manipulation of global and local explanations, where posthoc explanation techniques that cover up unfair black-box ML models are explored. Dishonest model producers can generate high-fidelity interpretable surrogate models to justify fairness, masking underlying unfairness. When automated AI systems dictate decisions, subjects are entitled not only to decision explanations but also to proof of their accuracy [156]. This demand intensifies scrutiny of the training data, potentially breaching the privacy rights of individuals from whom the data originated.According to [157], recent research on model explanations has faced criticism for neglecting actual usability considerations.

Similarly, another major interplay of explainability and transparency components is security component. Explaining and making transparent about nature of data, model architecture, training strategy, ML security methods, etc., creates backdoor for attacker at different level. In the research article [158], researchers have shown that based on the model explanation, it is possible to identify whether a particular set of data is included in the training set or not. The researchers named this type of attack an ”explanation-guided membership inference attack.” Another interesting paper [159] demonstrates how it is possible to extract the model only based on the gradient-based model explanations. This could create the biggest problem in multiple ways; for example, an adversary could reconstruct your sensitive model without authorizing it. The research community has termed such an attack a ”model extraction attack”, which is also an issue of intellectual property theft. The research article published in [160] further supports the argument that transparency opens a backdoor to attackers and makes it possible for them to not only construct the model partially or completely, but to reconstruct the training dataset. The author further emphasizes that explaining how the decision is made based on the ML algorithm from the perspective of complete transparency gives attackers the opportunity to design the attack to infer data or inject bogs into their existing frameworks or workflows. Consequently, the authors in [161] raised the question “Could an explainable model be inherently less secure?” Here, the authors argument is that the more one knows about the data and internal working mechanisms, the easier it is to deceive. In his recent paper [162] Adrian Weller mainly highlights the possible scenarios where transparency (one component of trustworthy AI) may cause harm. The author gives an example to explain how the higher level of transparency may lead to worse outcomes and also makes the model less fair [162]. Similarly, researchers from Stanford University published a paper claiming that the interpretation of neural networks is fragile [163]. To justify their claim, the authors have used two similar inputs with the same output but show that the explanation is completely different. The contribution of their work is to demonstrate (using the perturbation technique) how the interpretation of neural networks can be manipulated.

5.1.4 Fairness/robustness vs rest

ML model's fairness and robustness are two pillars of trustworthy AI. We have discussed more about these components in the trustworthy AI section 4. Because of similarity in nature, often similar types of solutions are used for dealing with fairness and robustness. Therefore, we have combined fairness and robustness under the same section. These fields have gained noticeable attention from researchers; however, their interactions have received less attention in the current scientific literature.

One of the approaches used for dealing with fairness (minority groups) and robustness (outlier data) is oversampling. The oversampling approach raises the issue of overfitting in ML models. An overfitted model creates serious issues for privacy, security, and explainability components of trustworthy AI. The study from [164] mentioned that overfitted models are highly prone to information leakage. Similarly, the overfitted models make it easier for attackers to design membership inference attacks to identify [165]. In addition, the explanation model faces the issue of generalization when it is based on an overfitted model. Data augmentation is a better approach for fairness and robustness compared to oversampling; however, it is still not a very reliable solution when it comes to highly imbalanced datasets. For instance, [166] demonstrates the risk of membership inference attacks through data augmentation.

Another advanced approach used by researchers in recent days is use of federated learning and distributed learning. These approaches are better than oversampling and data augmentation. However, these approaches are also prone to privacy issues at some level. For instance, recent articles [167, 168] mentioned that even though data is not directly exposed in FL and distributed learning, it is possible to extract sensitive information from the trained model.

Similarly, the interplay between fairness and privacy is investigated in [169]. The authors demonstrate that achieving fairness often comes at the expense of privacy. The privacy cost of fair models increases significantly for unprivileged subgroups, exacerbating the challenge of achieving fairness in biased training data.

In the case of robustness, the research community has explored the use of DP for enhancing ML robustness [74, 170]. When it comes to the use of the DP approach in ML, all the interactions we discussed in relation to the use of DP also apply here.

5.2 Current research gaps

The previous section on interplay demonstrates how the various components of trustworthiness are interdependent and interconnected. These interplays are just major representative cases that are visible openly. Nevertheless, there exists a significant degree of interaction among many constituents. The careful observation of current literature demonstrates the immaturity of understanding the connections and their effects among responsible parties. Another important observation we notice from current literature is the lack of collaboration between different experts.

Building the framework for trustworthy AI has gained huge attention from academic and industrial partners. There have already been some solutions available on the market for the evaluation of trustworthy AI. We have discussed different trustworthy AI evaluation frameworks in the trustworthy AI section of 4.2. However, one of the major issues is that these solutions do not meet the required level of standardness. Most of the trustworthy AI frameworks are in the immature phase [114]. For example, the Assessment List for Trustworthy AI (ALTAI), which is also presented in section 4.2, is an assessment method introduced by the EU itself. This assessment list is completely based on questions and answers. This assessment list is a good step in the right direction and provides good guidelines for building trustworthy applications. However, ALTAI has several weaknesses, such as [171], which criticizes that ALTAI does not meet the standard of origination-level development and does not consider the appropriate level of governance between different levels of organization that ought to be expected. Similarly, the appropriate level of interplay between different components is not properly addressed by the question.

One significant concern that has been identified is the lack of cohesion in the domain of trustworthy AI. As the topic has gained huge attention, different organizations and research groups have developed particular tools focusing on one particular problem. The fragmented nature of work lacks the ability to measure interconnection and its consequences. Building a trustworthy framework or application should be a collaborative effort between diverse experts. Achieving the desired outcome of establishing a reliable trustworthy AI framework through a fragmented approach is inherently unfeasible and unattainable.

5.3 Future action

Building a fully trustworthy AI application is a challenging task. Indeed, despite wide interest in trustworthy research, satisfactory solutions are still far from reach. We have already discussed current research gaps in the previous section. In this section, we would like to highlight the potential future action to bring trustworthy AI to the next level.

As we have emphasized a lot, the method used to achieve the goal of one of the components of trustworthy AI raises issues for the remaining components, which is one of the biggest issues. Therefore, careful analysis and research are crucial to figuring out the interplay between different components of trustworthy AI and how their side effects could be adequately minimized. One effective starting step could be to conduct a detailed study to identify all the possible interplay between different components of trustworthy AI. Such kind of study demands wide collaboration between researchers from academic and industrial fields, involvement from small to large companies, and expertise from legal, ethical, social, and technical teams. In order to establish a universally accepted framework for everyone, it is even necessary to have global collaboration. Such an approach of bringing expertise from diverse backgrounds to a single table is essential to analyzing and defining the road map for a universally accepted, trustworthy AI framework.

Similarly, another important future action is to view the development of trustworthy AI frameworks from a holistic perspective. Merging and combining all the fragmented solutions within the scope of trustworthy AI as a single solution is of paramount importance. Such a unified solution helps to achieve the goal of building trustworthy applications by considering all components and their interactions. In addition to this, building such a unified framework needs to be treated as a long-term goal. As technologies advance, the requirements and challenges change over a period of time. The solution that we consider effective at this moment may not be effective in the near future. Therefore, continuous collaboration, updates, and integration are necessary for a long period of

Table 3 Summary of the interplay between trustworthy AI components, highlighting some shortcomings when addressing different components simultaneously

	Trustworthy AI Components	Consequences & Shortcomings	Bib. sources
Privacy	Explainability	Poor and inconsistent explanation	[135–137]
	Accuracy	Reduce accuracy of model	[132, 133]
	Robustness	Reduce robustness of model	[139–141]
	Fairness	Reduce fairness of model	[138]
	Security	Increase risk of poisoning attacks and gradient masking	[140–145]
	Accountability	who is accountable for worst performance because of other solutions	[134]
Security	Explainability	Indirect negative impact on explainable model	[53, 147]
	Privacy	Leakage of membership information	[148]
	Fairness	Indirect negative impact on fairness	[138]
	Robustness	Indirect negative impact on robustness	[139–141]
	Performance	Degrade performance	[146]
Explainability/Transparency	Privacy	Information leakage	[136, 153, 155]
	Security	Makes easier to attacker to design dedicated attack, membership inference and poisoning attacks	[160–162]
Fairness/ Robustness	Privacy	Information leakage	[169]
	Explainability	Generalization issue in explainable model	[135]
	Security	membership inference and poisoning attacks	[167, 168]

time. Another potential future action could be to combine different frameworks built from different perspectives. For example, some trustworthy frameworks are built for technical people, while others focus on the administrative level. This will all increase the extra work required for people to know multiple frameworks. In addition to this, there is a high chance that the information may differ from framework to framework. For instance, a privacy score of 95% may have different meanings for technical reports and different meanings for legal teams.

Besides the trustworthy assessment framework, as recommended by researchers from [139], it could be a better approach to build ML models based on causality learning. The causality learning models not only improve their performance but also help to achieve better security, privacy, robustness, and fairness. From a privacy and security perspective, it will be very difficult to attack causality-based models as compared to normal features or correlation-based models.

6 Conclusions

In this paper, we have presented a thorough review of the recent research literature addressing ML security and privacy issues in connection with trustworthy AI. Our review shows that these topics have gained attention from both the research community and industrial sectors. However, the overall outcome of all this available literature still shows a lack of adequate solutions to address the main security and privacy issues in the context of trusted AI. Most of the solutions are still in the developing and testing phase and need further maturation and testing in different specific cases. For example, while FL is a promising new paradigm that might improve the privacy requirements of AI users in the future, there are still several challenges that need to be addressed efficiently, namely how to mitigate the computational expenses in data and algorithm sharing between the different parts of the FL framework, and how to avoid trade-offs with the accuracy of the algorithm, as well as possible new sources of bias and uncertainty.

As for research on reliable AI, it reveals that this field has captured the attention of governments and policy makers, in addition to researchers and industrial experts. Strict standards and regulations have been introduced at national and international levels. Therefore, the topic has become a hot topic in the field of AI. However, when studying related work in this field, we found that the research community has paid much more attention to the explanation part of the model, which is a component of trustworthy AI. To achieve a high level of trustworthiness, the research community should pay equal attention to other components.

In fact, even in the specific context of privacy and security, there is still a significant lack of unified solutions or benchmarks for assessing privacy, security or trustworthiness. Typically, AI research in these contexts is oriented toward solving specific problems. For example, we have shown the case of a proposed solution based on feature selection to improve the security of ML, while other works focus on enhancing the robustness of the model without considering any related privacy or security constraints. We believe that there is a need for a single framework containing multiple security solutions, which would allow users to use only that framework, and to run the model through different security tests and secure the model. The same is true for ML privacy-related solutions.

Due to their specificity, several approaches and models are still too limited or present considerable drawbacks. For example, so-called data noise [172] could be introduced, but it would also reflect on the possible bias of the model trained and tested on such data. An alternative would be to develop deep learning models, e.g. autoencoder [173], capable of masking the original data and sharing it with model developers in an FL framework. The developers could then send the “masked” prediction back to the data owner, who could invert the mask to obtain the “real” prediction. Other ways to protect data for use beyond the circle of ownership without losing the ability to be processed by modeling experts would be to develop models to produce surrogate data only from aggregated data. Recently, some work has been developed in these various directions [174].

Finally, there are two important open problems that merge the different aspects of reliable AI discussed in this paper, in particular security, privacy, explainability, and human trust. First, in order to develop reliable AI tools and algorithms and highly optimized security, privacy, reliability, trustworthiness and other requirements, it is necessary to develop evaluation tools to properly assess these features in the context of AI and, in particular, to quantify them.

Second, the assessment procedure for these different aspects of trustworthiness should not be carried out separately. While studying security, privacy, and trustworthy AI individually, we realized that there is a huge research gap in these fields. We believe that there are strong interconnections and dependencies among these three fields. Action taken to solve a specific problem in one domain may raise a problem in other domains. For example, if we use an ML model for credit card fraud detection, the transparency and explainability of a trustworthy AI share information related to, for example, data distribution, model architecture, important features, and the decision process, which could pose a major problem from a privacy perspective. Sharing information related to the model and data can help to filter out personal information from training data. In addition, an attacker can reconstruct the model based on those model explanations.

Therefore, instead of considering these three fields separately, they need to be viewed and developed as a unified solution. In particular, the assessment framework should consider the different measures simultaneously to explore possible trade-offs. For example, a very secure ML algorithm may have its explainability or accuracy compromised. Appropriately weighing the level of importance in meeting security or privacy requirements, along with the performance of the algorithm, depends on the specific context and objective related to the use of the algorithm. While such a multi-constraint optimization framework has already been at least partially addressed, for example, exploring Trustworthy AI based on distributed ledger technology [175] is still an open problem that will only be possible to address with interdisciplinary teams ranging from technical AI developers to social scientists and experts from the different fields in which AI is used.

Author Contributions Author contributions R.U was supervised by Prof. A.Y, Prof. P.L. and Dr. A.M while writing the survey, discussing structure and content. A.Y, Prof. P.L. and Dr. A.M did many rounds of rewriting and proof reading to finalize the manuscript.

Funding Open access funding provided by OsloMet - Oslo Metropolitan University

Data availability Data sharing is not applicable to this article as no datasets were generated or analyzed during the current study.

Declarations

Conflict of interest The authors of this article do not have any financial or non-financial interests to disclose that are relevant to the content. The authors also have no affiliations or involvement with any organization or entity that has a financial or non-financial interest in the subject matter or materials discussed in the manuscript. Furthermore, the authors have no financial or proprietary interests in any of the materials discussed in the article. As such, the authors declare no competing interests.

Ethical approval This research was funded by the Nordic Center for Sustainable and Trustworthy AI Research (NordSTAR) project. The project does not involve the use of human participants or animals. Therefore, ethical considerations related to human or animal subjects do not apply to this study.

References

1. Oseni, A., Moustafa, N., Janicke, H., Liu, P., Tari, Z., Vasilakos, A.: Security and Privacy for Artificial Intelligence: Opportunities and Challenges. arXiv preprint: arXiv:2102.04661 (2021)
2. Finlayson, S.G., Bowers, J.D., Ito, J., Zittrain, J.L., Beam, A.L., Kohane, I.S.: Adversarial attacks on medical machine learning. Science **363**(6433), 1287 (2019)
3. Peters, D., Vold, K., Robinson, D., Calvo, R.A.: Responsible AI-two frameworks for ethical design practice. IEEE Trans. Technol. Soc. **1**(1), 34 (2020)
4. El Naqa, I., Murphy, M.J.: in What is Machine Learning? Machine Learning in Radiation Oncology (Springer, 2015), pp. 3–11
5. Kubat, M., Kubat, An Introduction to Machine Learning: An Introduction to Machine Learning, vol. 2 (Springer, 2017)
6. Mothukuri, V., Parizi, R.M., Pouriyeh, S., Huang, Y., Dehghantanha, A., Srivastava, G.: A survey on security and privacy of federated learning. Future Generat. Comput. Syst. **115**, 619 (2021). https://doi.org/10.1016/j.future.2020.10.007.www.sciencedirect.com/science/article/pii/S0167739X20329848
7. High-Level Expert Group on Artificial Intelligence of the European Commission. Ethics guidelines for trustworthy ai. high-level expert group on artificial intelligence (2019)
8. Liu H., Wang, Y., Fan, W., Liu, X., Li, Y., Jain, S., Liu, Y., Jain, A.K., Tang, J.: arXiv preprint arXiv:2107.06641 (2021)
9. Pitropakis, N., Panaousis, E., Giannetsos, T., Anastasiadis, E., Loukas, G.: A taxonomy and survey of attacks against machine learning. Comput. Sci. Rev. **34**, 100199 (2019)
10. Papernot, N., McDaniel, P., Sinha, A., Wellman, M.:Towards the Science of Security and Privacy in Machine Learning. arXiv preprint: arXiv:1611.03814 (2016)
11. Barreno, M., Nelson, B., Joseph, A.D., Tygar, J.D.: The security of machine learning. Mach. Learn. **81**(2), 121 (2010)
12. Liu, Q., Li, P., Zhao, W., Cai, W., Yu, S., Leung, V.C.: A survey on security threats and defensive techniques of machine learning: a data driven view. IEEE Access **6**, 12103 (2018)
13. Newsome, J., Karp, B., Song, D.: Thwarting signature learning by training maliciously. In: International workshop on recent advances in intrusion detection paragraph: thwarting signature learning by training maliciously (Springer, 2006), pp. 81–105
14. Burkard, C., Lagesse, B.: Analysis of causative attacks against svms learning from data streams. In: Proceedings of the 3rd ACM on International Workshop on Security And Privacy Analytics (2017), pp. 31–36
15. Shi, Y., Sagduyu, Y.E., Evasion and causative attacks with adversarial deep learning. In: MILCOM 2017–2017 IEEE Military Communications Conference (MILCOM) (IEEE, 2017), pp. 243–248
16. Sihag, S., Tajer, A.: Secure estimation under causative attacks. IEEE Trans. Inf. Theory **66**(8), 5145 (2020)
17. Mozaffari-Kermani, M., Sur-Kolay, S., Raghunathan, A., Jha, N.K.: Systematic poisoning attacks on and defenses for machine learning in healthcare. IEEE J. Biomed Health Informatics **19**(6), 1893 (2014)
18. Baracaldo, N., Chen, B., Ludwig, H., Safavi, J.A.: In Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security (2017), pp. 103–110
19. Sagduyu, Y.E., Shi, Y., Erpek, T.: Adversarial deep learning for over-the-air spectrum poisoning attacks. IEEE Trans. Mobile Comput. **20**(2), 306 (2019)
20. Seth,i T.S., Kantardzic, M.M.:"Security theater": on the vulnerability of classifiers to exploratory attacks Data driven exploratory attacks on black box classifiers in adversarial domains. Neurocomputing **289**, 129 (2018)

21. Sethi, T.S., Kantardzic, M., Ryu, J.W.: in Pacific-Asia Workshop on Intelligence and Security Informatics (Springer, 2017), pp. 49–63
22. Lin, X., Zhou, C., Yang, H., Wu, H. Wang, Y. Cao, B. Wang, Exploratory adversarial attacks on graph neural networks. In: 2020 IEEE International Conference on Data Mining (ICDM) (IEEE, 2020), pp. 1136–1141
23. Shi, Y., Sagduyu, Y., Grushin, A.: How to steal a machine learning classifier with deep learning. In: 2017 IEEE International Symposium on Technologies for Homeland Security (HST) (IEEE, 2017), pp. 1–5
24. D. Shu, N.O. Leslie, C.A. Kamhoua, C.S. Tucker. In: Proceedings of the 2nd ACM Workshop on Wireless Security and Machine Learning (2020), pp. 1–6
25. Ji, Y., Bowman, B., Huang, H.H.: Generative adversarial attacks against intrusion detection systems using active learning. In: 2019 IEEE International Conference on Cognitive Computing (ICCC) (IEEE, 2019), pp. 1–9
26. Fazelnia, M., Khokhlov, I., Mirakhorli, M.: Attacks, Defenses, and Tools: A Framework to Facilitate Robust AI/ML Systems. arXiv preprint: arXiv:2202.09465 (2022)
27. Clark, G., Doran, M., Glisson, W.: In 2018 17th IEEE International Conference on Trust, Security and Privacy in Computing And Communications/12th IEEE International Conference on Big Data Science and Engineering (TrustCom/BigDataSE) (IEEE, 2018), pp. 516–521
28. Papernot, N., McDaniel, P., Goodfellow, I., Jha, S., Celik, Z.B., Swami A.: In Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security (2017), pp. 506–519
29. Gao, L., Cheng, Y., Zhang, Q., Xu, X., Song, J.: Feature Space Targeted Attacks by Statistic Alignment. arXiv preprint: arXiv:2105.11645 (2021)
30. Newaz, A.I., Haque, N.I., Sikder, A.K., Rahman, M.A., Uluagac, A.S.: in GLOBECOM 2020–2020 IEEE Global Communications Conference (IEEE, 2020), pp. 1–6
31. Tian, J., Wang, B., Li, J., Wang, Z., Ma, B., Ozay, M.: Exploring targeted and stealthy false data injection attacks via adversarial machine learning. IEEE Internet Things J (2022)
32. Kozlowski, M., Ksiezopolski, B.: A new method of testing machine learning models of detection for targeted DDoS attacks. In: SECRYPT (2021), pp. 728–733
33. Ughi, G., Abrol, V., Tanner, J.: An empirical study of derivative-free-optimization algorithms for targeted black-box attacks in deep neural networks. Opt. Eng., pp. 1–28 (2021)
34. Hong, S., Chandrasekaran, V., Kaya, Y., Dumitraş, T., Papernot, N.: On the effectiveness of mitigating data poisoning attacks with gradient shaping. arXiv preprint: arXiv:2002.11497 (2020)
35. Rawal, A., Rawat, D., Sadler, B.M.: in Artificial Intelligence and Machine Learning for Multi-Domain Operations Applications III, vol. 11746 (International Society for Optics and Photonics, 2021), vol. 11746, p. 117462Q
36. Duddu, V.: A survey of adversarial machine learning in cyber warfare. Defence Sci. J. **68**(4), 356 (2018)
37. Davoodi, M., Moslemi, R., Song, W., Velni, J.M.: A fog-based approach to secure smart grids against data integrity attacks. In: 2020 IEEE Power and Energy Society Innovative Smart Grid Technologies Conference (ISGT) (IEEE, 2020), pp. 1–5
38. Badrinath Krishna, V., Weaver, G.A., Sanders, W.H.: PCA-based method for detecting integrity attacks on advanced metering infrastructure. In: International Conference on Quantitative Evaluation of Systems (Springer, 2015), pp. 70–85
39. Auernhammer, K., Kolagari, R.T., Zoppelt, M.: Attacks on machine learning: lurking danger for accountability. In: SafeAI@ AAAI (2019)
40. Almalawi, A., Yu, X., Tari, Z., Fahad, A., Khalil, I.: An unsupervised anomaly-based detection approach for integrity attacks on SCADA systems. Comput. Security **46**, 94 (2014)
41. Newell, A., Potharaju, R., Xiang, L., Nita-Rotaru, C.: In: Proceedings of the 2014 Workshop on Artificial Intelligent and Security Workshop (2014), pp. 83–93
42. Duan, J., Chow, M.Y.: A resilient consensus-based distributed energy management algorithm against data integrity attacks. IEEE Trans. Smart Grid **10**(5), 4729 (2018)
43. Yang, X., Zhang, X., Lin, J., Yu, W., Zhao, p. : In: 2016 25th International Conference on Computer Communication and Networks (ICCCN) (IEEE, 2016), pp. 1–9
44. Farraj, A., Hammad, E., Kundur, D.: A distributed control paradigm for smart grid to address attacks on data integrity and availability IEEE Transactions on Signal and Information Processing over. Networks **4**(1), 70 (2017)
45. Yu, Y., Liu, X., Chen, Z.: In: Proceedings of the 2nd International Conference on Computer Science and Application Engineering (2018), pp. 1–7
46. Nelson, B., Barreno, M., Chi, F.J., Joseph, A.D., Rubinstein, B.I., Saini, U., Sutton, C., Tygar, J.D., Xia, K.: Exploiting machine learning to subvert your spam filter. LEET **8**(1–9), 16 (2008)
47. Jagielski, M., Oprea, A., Biggio, B., Liu, C., Nita-Rotaru, C., Li, B.: In: 2018 IEEE Symposium on Security and Privacy (SP) (IEEE, 2018), pp. 19–35
48. Ford, V., Siraj, A.: In: Proceedings of the 27th International Conference on Computer Applications in Industry and Engineering, vol. 118 (IEEE Xplore Kota Kinabalu, Malaysia, 2014), vol. 118
49. Papernot, N., McDaniel, P., Sinha, A., Wellman, M.P.: In: 2018 IEEE European Symposium on Security and Privacy (EuroS&P) (IEEE, 2018), pp. 399–414
50. Shumailov, I., Shumaylov, Z., Kazhdan, D., Zhao, Y., Papernot, N., Erdogdu, M.A., Anderson, R.: Manipulating sgd with data ordering attacks. Adv. Neural Inf. Process. Syst. **34** (2021)
51. Barreno, M., Nelson, B., Sears, R., Joseph, A.D., Tygar, J.D.: In: Proceedings of the 2006 ACM Symposium on Information, computer and communications security (2006), pp. 16–25
52. Imam, N.H., Vassilakis, V.G.: A survey of attacks against twitter spam detectors in an adversarial environment. Robotics **8**(3), 50 (2019)
53. Rigaki, M., Garcia, S.: A survey of privacy attacks in machine learning. arXiv preprint arXiv:2007.07646 (2020)
54. Sherman, M.: Influence attacks on machine learning (2020). https://ai4.io/blog/2020/04/01/influence-attacks-on-machine-learning/
55. Yuan, X., He, P., Zhu, Q., Li, X.: Adversarial examples: Attacks and defenses for deep learning. IEEE Trans. Neural Netw. Learn. Syst. **30**(9), 2805 (2019)
56. Sagar, R., Jhaveri, R., Borrego, C.: Applications in security and evasions in machine learning: a survey. Electronics **9**(1), 97 (2020)
57. Peng, J., Chan, P.P.: In: 2013 International Conference on Machine Learning and Cybernetics, vol. 2 (IEEE, 2013), vol. 2, pp. 610–614
58. Siddiqi, A.: Adversarial Security Attacks and Perturbations on Machine Learning and Deep Learning Methods. arXiv preprint: arXiv:1907.07291 (2019)
59. Rathore, P., Basak, A., Nistala, S.H., Runkana, V.: In: 2020 International Joint Conference on Neural Networks (IJCNN) (IEEE, 2020), pp. 1–8
60. Akhtar, N., Mian, A.: Threat of adversarial attacks on deep learning in computer vision: a survey IEEE. Access **6**, 14410 (2018)
61. Goodfellow, I.J., Shlens, J., Szegedy, C.: arXiv preprint arXiv:1412.6572 (2014)
62. Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D.I. Goodfellow, R. Fergus. arXiv preprint: arXiv:1312.6199 (2013)

63. Papernot, N., McDaniel, P., Wu, X., .Jha, S., Swami, A.: In: 2016 IEEE Symposium on Security Pnd privacy (SP) (IEEE, 2016), pp. 582–597
64. Roden, B., Lusher, D., Spurling, T.H., Simpson, G.W., Klein, T., Brailly, J., Hogan, B.: Avoiding GIGO: Learnings from Data Collection in Innovation Research Social Networks (2020)
65. Koh, P.W., Steinhardt, J., Liang, P.: Stronger data poisoning attacks break data sanitization defenses. Mach. Learn. **111**(1), 1 (2022)
66. Peng, R., Xiao, H., Guo, J., Lin, C.: Defending a parallel system against a strategic attacker with redundancy, protection and disinformation. Reliabil. Eng. Syst. Safety **193**, 106651 (2020)
67. Baloun, K., CHANG, K., Holmes, M.: Disinformation Defense of AI Inference UNIVERSITY OF CALIFORNIA–BERKELEY (2019)
68. Sweeney, L.: k-Anonymity: a model for protecting privacy. Int. J. Uncertain. Fuzziness Knowl. Based Syst. **10**(05), 557 (2002)
69. El Emam, K., Dankar, F.K.: Protecting privacy using k-anonymity. J. Am. Med. Informat. Assoc. **15**(5), 627 (2008)
70. Globerson, A., Roweis, S.: In: Proceedings of the 23rd International Conference on Machine Learning (2006), pp. 353–360
71. Iqbal, R.A.: Using Feature Weights to Improve Performance of Neural Networks. arXiv preprint: arXiv:1101.4918 (2011)
72. Dhillon, G.S., Azizzadenesheli, K., Lipton, Z.C., Bernstein, J., Kossaifi, J. , Khanna, A., Anandkumar, A.: Stochastic activation pruning for robust adversarial defense. arXiv preprint: arXiv:1803.01442 (2018)
73. Xie, C., Wang, J., Zhang, Z., Ren, Z., Yuille, A.: Mitigating adversarial effects through randomization. arXiv preprint: arXiv:1711.01991 (2017)
74. Lecuyer, M., Atlidakis, V., Geambasu, R., Hsu, D., Jana, S.: In: 2019 IEEE symposium on security and privacy (SP) (IEEE, 2019), pp. 656–672
75. Wang, B., Shi, Z., Osher, S.: Resnets ensemble via the feynman-kac formalism to improve natural and robust accuracies. Adv. Neural Inf. Process. Syst. **32** (2019)
76. Pinot, R., Meunier, L., Araujo, A., Kashima, H., Yger, F., Gouy-Pailler, C., Atif, J. : Theoretical evidence for adversarial robustness through randomization. Adv. Neural Inf. Process. Syst. **32** (2019)
77. Pinot, R., Ettedgui, R., Rizk, G., Chevaleyre, Y., Atif, J.:Randomization matters how to defend against strong adversarial attacks. In: International Conference on Machine Learning (PMLR, 2020), pp. 7717–7727
78. Wang, X., Li, J., Kuang, X., Tan, Y.A., Li, J.: The security of machine learning in an adversarial setting: a survey. J. Parall. Distribut. Comput. **130**, 12 (2019)
79. Yi, X., Paulet, R., Bertino, E.: Differential privacy and machine learning: a survey and review. In: Homomorphic Encryption and Applications (Springer, 2014), pp. 27–46
80. Ji, Z., Lipton, Z.C., Elkan, C.: Differential privacy and machine learning: a survey and review. arXiv preprint: arXiv:1412.7584 (2014)
81. Gray, R.: Vector quantization. IEEE Assp. Mag. **1**(2), 4 (1984)
82. Pieprzyk, J., Sadeghiyan, B., Design of Hashing Algorithms (Springer, 1993)
83. Xu, F., Peng, J., Xiang, J., Zha, D.: In: 2019 IEEE SmartWorld, Ubiquitous Intelligence and Computing, Advanced and Trusted Computing, Scalable Computing and Communications, pp. 1237–1242. Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI) (IEEE, Cloud & Big Data Computing (2019)
84. McMahan, B., Moore, E., Ramage, D., Hampson, S., y Arcas B.A.: In: Artificial Intelligence and Statistics (PMLR, 2017), pp. 1273–1282
85. Chen, S., Xue, D., Chuai, G., Yang, Q., Liu, Q.: FL-QSAR: a federated learning-based QSAR prototype for collaborative drug discovery. Bioinformatics **36**(22–23), 5492 (2020)
86. Yang, Q., Liu, Y., Chen, T., Tong, Y.: Federated machine learning: concept and applications. ACM Trans. Intell. Syst. Technol. (TIST) **10**(2), 1 (2019)
87. Mugunthan, V., Polychroniadou, A., Byrd, D., Balch, T.H.: in Proceedings of the NeurIPS 2019 Workshop on Robust AI in Financial Services (MIT Press Cambridge, MA, USA, 2019), pp. 1–9
88. Wei, K., Li, J., Ding, M., Ma, C., Yang, H.H., Farokhi, F., Jin, S., Quek, T.Q., Poor, H.V.: Federated learning with differential privacy: algorithms and performance analysis. IEEE Trans. Inf. Forensics Security **15**, 3454 (2020)
89. Truex, S., Baracaldo, N., Anwar, A., Steinke, T., Ludwig, H., Zhang, R., Zhou, Y.: A hybrid approach to privacy-preserving federated learning. In: Proceedings of the 12th ACM Workshop on Artificial Intelligence and Security (2019), pp. 1–11
90. Liu, X., Li, H., Xu, G., Lu, R., He, M.: Adaptive privacy-preserving federated learning. Peer Peer Networ. Appl. **13**(6), 2356 (2020)
91. Gharibi, M., Rao, P.: In: 2020 IEEE Applied Imagery Pattern Recognition Workshop (AIPR) (IEEE, 2020), pp. 1–5
92. Bag, S.: Federated Learning—A Beginners Guide Download (2021). https://www.analyticsvidhya.com/blog/2021/05/federated-learning-a-beginners-guide/
93. Nasr, M., Shokri, R., Houmansadr, A.:Comprehensive privacy analysis of deep learning: stand-alone and federated learning under passive and active white-box inference attacks. In: Proceedings of the 2019 IEEE Symposium on Security and Privacy (SP) (2018), pp. 1–15
94. Choudhury, O., Gkoulalas-Divanis, A., Salonidis, T., Sylla, I., Park, Y. , Hsu, G., Das, A.: arXiv preprint: arXiv:2002.09096 (2020)
95. Amiri, M.M., Gunduz, D., Kulkarni, S.R., Poor, H.V.:Federated learning with quantized global model update: arXiv preprint: arXiv:2006.10672 (2020)
96. Lee, J., Sun, J., Wang, F., Wang, S., Jun, C.H., Jiang, X.: Privacy-preserving patient similarity learning in a federated environment: development and analysis. JMIR Medi. Informat. **6**(2), e7744 (2018)
97. Konečnỳ, J., McMahan, H.B., Yu, F.X., Richtárik, P., Suresh, A.T., Bacon, D.: arXiv preprint: arXiv:1610.05492 (2016)
98. G. Boesch. An introduction to federated learning: Challenges and applications technology-wallpaper-deep-learning (2021). https://viso.ai/deep-learning/federated-learning/
99. Rahman, K.J., Ahmed, F., Akhter, N., Hasan, M., Amin, R., Aziz, K.E., Islam, A.M., Mukta, M.S.H., Islam, A.N.: Challenges, applications and design aspects of federated learning: a survey. IEEE Access **9**, 124682 (2021)
100. Nishio, T., Yonetani, R.: Client selection for federated learning with heterogeneous resources in mobile edge. In: ICC 2019-2019 IEEE International Conference on Communications (ICC) (IEEE, 2019), pp. 1–7
101. Smith V., Chiang, C.K. , Sanjabi, M., Talwalkar, A.: arXiv preprint: arXiv:1705.10467 (2017)
102. Li, T., Sahu, A.K., Talwalkar, A., Smith, V.: Federated learning: challenges, methods, and future directions. IEEE Signal Process. Mag. **37**(3), 50 (2020)
103. Rieke, N., Hancox, J., Li, W., Milletari, F., Roth, H.R., Albarqouni, S., Bakas, S., Galtier, M.N., Landman, B.A., Maier-Hein, K., et al.: The future of digital health with federated learning. NPJ Digital Med. **3**(1), 1 (2020)
104. Hard, A., Rao, K., Mathews, R., Ramaswamy, S., Beaufays, F., Augenstein, S., Eichner, H., Kiddon, C., Ramage, D.: Fed-

erated learning for mobile keyboard prediction. arXiv preprint: arXiv:1811.03604 (2018)

105. Sarma, K.V., Harmon, S., Sanford, T., Roth, H.R., Xu, Z., Tetreault, J., Xu, D., Flores, M.G., Raman, A.G., Kulkarni, R., et al.: Federated learning improves site performance in multicenter deep learning without data sharing. J. Am. Med. Informat. Assoc. **28**(6), 1259 (2021)

106. NIST. trustworthiness (2023). https://csrc.nist.gov/glossary/term/trustworthiness#::text=trustworthiness

107. Adadi, A., Berrada, M.: Peeking inside the black-box: a survey on explainable artificial intelligence. IEEE Access **6**, 52138 (2018)

108. ML, T.: Trustworthy ml initiative (2022). https://www.trustworthyml.org/

109. Marino, D.L., Wickramasinghe, C.S., Manic, M.: An adversarial approach for explainable ai in intrusion detection systems. In: IECON 2018-44th Annual Conference of the IEEE Industrial Electronics Society (IEEE, 2018), pp. 3237–3243

110. Liu, H., Wang, Y., Fan, W., Liu, X., Li, Y., Jain, S., Liu, Y., Jain, A., Tang, J.: Trustworthy ai: a computational perspective. ACM Trans. Intell. Syst. Technol. **14**(1), 1 (2022)

111. Bærøe, K., Miyata-Sturm, A., Henden, E.: How to achieve trustworthy artificial intelligence for health Bulletin of World Hearth. Organ **98**, 257 (2020)

112. Chandler, C., Foltz, P.W., Elvevåg, B.: Using machine learning in psychiatry: the need to establish a framework that nurtures trustworthiness. Schizophrenia Bull. **46**(1), 11 (2020)

113. Khademi, A., Honavar, V.: Algorithmic bias in recidivism prediction: a causal perspective (student abstract). In: Proceedings of the AAAI Conference on Artificial Intelligence , 10, pp. 13,839–13,840 (2020)

114. Li, B., Qi, P., Liu, B., Di, S., Liu, J., Pei, J., Yi, J., Zhou, B.: Trustworthy AI: From principles to practices. ACM Comput. Surv. **55**(9), 1 (2023)

115. Ala-Pietilä, P., Bonnet, Y., Bergmann, U., Bielikova, M., Bonefeld-Dahl, C., Bauer, W., Bouarfa, L., Chatila, R., Coeckelbergh, M., Dignum, V.: et al., The Assessment List for Trustworthy Artificial Intelligence (ALTAI) (European Commission, 2020)

116. Nicolae, M.I., Sinn, M., Tran, M.N., Buesser, B., Rawat, A., Wistuba, M., Zantedeschi, V., Baracaldo, N., Chen, B., Ludwig, H., Molloy, I., Edwards, B.: Adversarial Robustness Toolbox v1.2.0, CoRR **1807.01069** (2018). https://arxiv.org/pdf/1807.01069

117. Nicolae, M.I., Sinn, M., Tran, M.N., Buesser, B., Rawat, A., Wistuba, M., Zantedeschi, V., Baracaldo, N., Chen, B., Ludwig, H.: et al., Adversarial Robustness Toolbox v1. 0.0. arXiv preprint arXiv:1807.01069 (2018)

118. Goldsteen, A., Farkash, A., Moffie, M., Shmelkin, R.: Applying artificial intelligence privacy technology in the healthcare domain. In: Challenges of Trustable AI and Added-Value on Health (IOS Press, 2022), pp. 121–122

119. Bellamy, R.K., Dey, K., Hind, M., Hoffman, S.C., Houde, S., Kannan, K., Lohia, P., Martino, J., Mehta, S., Mojsilovic, A.: et al., AI Fairness 360: An Extensible Toolkit for Detecting. Understanding, and Mitigating Unwanted Algorithmic Bias (2018)

120. Arnold, M., Bellamy, R.K., Hind, M., Houde, S., Mehta, S., Mojsilović, A., Nair, R., Ramamurthy, K.N., Olteanu, A., Piorkowski, D., et al.: FactSheets: Increasing trust in AI services through supplier's declarations of conformity. IBM J. Res. Develop. **63**(4/5), 6 (2019)

121. Baracaldo, N., Anwar, A., Purcell, M., Rawat, A., Sinn, M., Altakrouri, B., Balta, D., Sellami, M., Kuhn, P., Schopp, U.: et al., Towards an accountable and reproducible federated learning: a FactSheets approach, arXiv preprint: arXiv:2202.12443 (2022)

122. Ghosh, S., Liao, Q.V., Ramamurthy, K.N., Navratil, J., Sattigeri, P., Varshney, K.R., Zhang, Y.: Uncertainty quantification 360: A holistic toolkit for quantifying and communicating the uncertainty of ai. arXiv preprint: arXiv:2106.01410 (2021)

123. Lundberg, S.M., Lee, S.I.: A unified approach to interpreting model predictions. Adv. Neural Inf. Process. Syst. **30** (2017)

124. Ribeiro, M.T., Singh, S., Guestrin, C.: "Why should i trust you?" Explaining the predictions of any classifier. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (2016), pp. 1135–1144

125. Arya, V., Bellamy, R.K., Chen, P.Y., Dhurandhar, A., Hind, M., Hoffman, S.C., Houde, S., Liao, Q.V., Luss, R., Mojsilovic, A., et al.: AI Explainability 360: An Extensible Toolkit for Understanding Data and Machine Learning Models. J. Mach. Learn. Res. **21**(130), 1 (2020)

126. Arya, V., Bellamy, R.K., Chen, P.Y., Dhurandhar, A., Hind, M., Hoffman, S.C., Houde, S., Liao, Q.V., Luss, R., Mojsilović, A.: et al., AI Explainability 360 Toolkit. In: Proceedings of the 3rd ACM India Joint International Conference on Data Science and Management of Data (8th ACM IKDD CODS & 26th COMAD) (2021), pp. 376–379

127. Varshney, K.R.: XRDS: Crossroads, the ACM magazine for students. Trustworthy Mach. Learn. Artif. Intell. **25**(3), 26 (2019)

128. Viganò, L., Magazzeni, D.: Explainable security. In: 2020 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW) (IEEE, 2020), pp. 293–300

129. Gunning, D., Aha, D.: DARPA's explainable artificial intelligence (XAI) program. AI Mag. **40**(2), 44 (2019)

130. Patel, N., Shokri, R., Zick, Y.: Model explanations with differential privacy. arXiv preprint: arXiv:2006.09129 (2020)

131. Franco, D., Oneto, L., Navarin, N., Anguita, D.: Toward learning trustworthily from data combining privacy, fairness, and explainability: an application to face recognition. Entropy **23**(8), 1047 (2021)

132. Bassily, R., Smith, A., Thakurta, A.: Private empirical risk minimization: Efficient algorithms and tight error bounds. In: 2014 IEEE 55th Annual Symposium on Foundations of Computer Science (IEEE, 2014), pp. 464–473

133. Wang, D., Ye, M., Xu, J.: Differentially private empirical risk minimization revisited: Faster and more general. Adv. Neural Inf. Process. Syst. **30** (2017)

134. Cooper, A.F., Moss, E., Laufer, B., Nissenbaum, H.: Accountability in an algorithmic society: relationality, responsibility, and robustness in machine learning. In: Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency (2022), pp. 864–876

135. Saifullah, S., Mercier, D., Lucieri, A., Dengel, A., Ahmed, S.: Privacy meets explainability: A comprehensive impact benchmark. arXiv preprint: arXiv:2211.04110 (2022)

136. Patel, N., Shokri, R., Zick, Y.: Model explanations with differential privacy. In: Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency (2022), pp. 1895–1904

137. Naidu, R., Priyanshu, A., Kumar, A., Kotti, S., Wang, H., Mireshghallah, F.:When differential privacy meets interpretability: a case study. arXiv preprint: arXiv:2106.13203 (2021)

138. Bagdasaryan, E., Poursaeed, O., Shmatikov, V.: Differential privacy has disparate impact on model accuracy. Adv. Neural Inf. Process. Syst. **32** (2019)

139. Gittens, A., Yener, B., Yung, M.: An adversarial perspective on accuracy, robustness, fairness, and privacy: multilateral-tradeoffs in trustworthy ML. IEEE Access **10**, 120850 (2022)

140. Tursynbek, N., Petiushko, A., Oseledets, I.: Robustness threats of differential privacy. arXiv preprint: arXiv:2012.07828 (2020)

141. Boenisch, F., Sperl, P., Böttinger, K.: Gradient masking and the underestimated robustness threats of differential privacy in deep learning. arXiv preprint: arXiv:2105.07985 (2021)

142. Cheu, A., Smith, A., Ullman, J.: Manipulation attacks in local differential privacy. In: 2021 IEEE Symposium on Security and Privacy (SP) (IEEE, 2021), pp. 883–900

143. Giraldo, J., Cardenas, A., Kantarcioglu, M., Katz, J.: Adversarial classification under differential privacy. In Network and Distributed Systems Security (NDSS) Symposium 2020 (2020)
144. Hossain, M.T., Islam, S., Badsha, S., Shen, H.: Desmp: Differential privacy-exploited stealthy model poisoning attacks in federated learning. In: 2021 17th International Conference on Mobility, Sensing and Networking (MSN) (IEEE, 2021), pp. 167–174
145. Wu, Y., Cao, X., Jia, J., Gong, N.Z.: Poisoning attacks to local differential privacy protocols for {Key-Value} data. In: 31st USENIX Security Symposium (USENIX Security 22) (2022), pp. 519–536
146. Xue, M., Yuan, C., Wu, H., Zhang, Y., Liu, W.: Machine learning security: threats, countermeasures, and evaluations. IEEE Access **8**, 74720 (2020)
147. Tramèr, F., Zhang, F.: A. Juels, M.K. Reiter, T. Ristenpart, Stealing machine learning models via prediction {APIs}. In 25th USENIX Security Symposium (USENIX Security 16) (2016), pp. 601–618
148. Song, L., Shokri, R., Mittal, P.: In: Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security (2019), pp. 241–257
149. Strobel, M., Shokri, R.: Data privacy and trustworthy machine learning. IEEE Security Privacy **20**(5), 44 (2022)
150. Alamäki, A., Mäki, M., Ratnayake, R.: Privacy concern, data quality and trustworthiness of AI-analytics. In: Proceedings of Fake Intelligence Online Summit 2019 (2019)
151. Ancona, M., Ceolini, E., Öztireli, C., Gross, M.: arXiv preprint: arXiv:1711.06104 (2017)
152. Bach, S., Binder, A., Montavon, G., Klauschen, F., Müller, K.R., Samek, W.: On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. PloS One **10**(7), e0130140 (2015)
153. Shokri, R., Strobel, M., Zick, Y.: On the privacy risks of model explanations. In: Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society (2021), pp. 231–241
154. Aïvodji, U., Bolot, A., Gambs, S.: Model extraction from counterfactual explanations. arXiv preprint: arXiv:2009.01884 (2020)
155. Aïvodji, U., Arai, H., O. Fortineau, H., Gambs, S., Hara, S., Tapp, A.: In: International Conference on Machine Learning (PMLR, 2019), pp. 161–170
156. Grant, T.D., Wischik, D.J.: Show us the data: Privacy, explainability, and why the law can't have both. Geo. Wash. L. Rev. **88**, 1350 (2020)
157. Kaur, H., Nori, H., Jenkins, S., Caruana, R., Wallach, H., Wortman Vaughan,J.: Interpreting interpretability: understanding data scientists' use of interpretability tools for machine learning. In: Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (2020), pp. 1–14
158. Quan, P., Chakraborty, S., Jeyakumar, J.V., Srivastava, M.: arXiv preprint: arXiv:2206.14004 (2022)
159. Milli, S., Schmidt, L., Dragan, A.D., Hardt, M.: Model reconstruction from model explanations. In: Proceedings of the Conference on Fairness, Accountability, and Transparency (2019), pp. 1–9
160. Chaudhury, D.: Fighting the Risks Associated with Transparency of ai Models (2020). https://enterprisetalk.com/featured/fighting-the-risks-associated-with-transparency-of-ai-models/
161. jaoka, A.: Could an Explainable Model be Inherently Less Secure?, Could an Explainable Model be Inherently Less Secure? (2022). https://www.datasciencecentral.com/could-an-explainable-model-be-inherently-less-secure/
162. Weller, A.: Transparency: motivations and challenges. In Explainable AI: Interpreting, Explaining and Visualizing Deep Learning (Springer, 2019), pp. 23–40
163. Ghorbani, A., Abid, A, Zou, J.: In: Proceedings of the AAAI Conference on Artificial Intelligence (2019), 01, pp. 3681–3688
164. Song, C., Raghunathan, A.: In Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security (2020), pp. 377–390
165. Song, L., Shokri, R., Mittal, P.: In: 2019 IEEE Security and Privacy Workshops (SPW) (IEEE, 2019), pp. 50–56
166. Liu, Y., Jiang, P., Zhu, L.: IEEE Transactions on Information Forensics and Security (2023)
167. So, J., Ali, R.E., Güler, B., Jiao, J., Avestimehr, A.S.: In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 37 (2023), pp. 9864–9873
168. Shao, J., Li, Z., Sun, W., Zhou, T., Sun, Y., Liu, L., Lin,Z., Zhang, J.: A survey of what to share in federated learning: perspectives on model utility, privacy leakage, and communication efficiency. arXiv preprint: arXiv:2307.10655 (2023)
169. Chang, H., Shokri, R.: In: 2021 IEEE European Symposium on Security and Privacy (EuroS&P) (IEEE, 2021), pp. 292–303
170. Chhor, J., Sentenac, F.: In: International Conference on Algorithmic Learning Theory (PMLR, 2023), pp. 411–446
171. Radclyffe, C., Ribeiro, M., Wortham, R.H.: The assessment list for trustworthy artificial intelligence: a review and recommendations. Front. Artif. Intell. **6**, 1020592 (2023)
172. Jain, S.K., Kesswani, N. : A noise-based privacy preserving model for Internet of Things. Complex Intell. Syst., pp. 1–25 (2021)
173. Bank, D., Koenigstein, N., Giryes, R.: arXiv preprint: arXiv:2003.05991 (2020)
174. Small, M., Nakamura, T., Luo, X.: Nonlinear Phenomena Research Perspectives, pp. 55–81 (2007)
175. Thiebes, S., Lins, S., Sunyaev, A.: Trustworthy artificial intelligence. Electronic Markets **31**, 447 (2021)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.