Patterns

CellPress
OPEN ACCESS

Review

Privacy preservation for federated learning in health care

Sarthak Pati,[1,2,15] Sourav Kumar,[3,15] Amokh Varma,[3,15] Brandon Edwards,[4,15] Charles Lu,[3,5] Liangqiong Qu,[6] Justin J. Wang,[7] Anantharaman Lakshminarayanan,[8] Shih-han Wang,[4] Micah J. Sheller,[4] Ken Chang,[9] Praveer Singh,[10] Daniel L. Rubin,[7] Jayashree Kalpathy-Cramer,[10,16] and Spyridon Bakas[1,2,11,12,13,14,16,*]

[1]Center for Federated Learning in Medicine, Indiana University, Indianapolis, IN, USA
[2]Division of Computational Pathology, Department of Pathology and Laboratory Medicine, Indiana University School of Medicine, Indianapolis, IN, USA
[3]Department of Radiology, Athinoula A. Martinos Center for Biomedical Imaging, Massachusetts General Hospital, Boston, MA, USA
[4]Intel Corporation, Santa Clara, CA, USA
[5]Center for Clinical Data Science, Massachusetts General Hospital and Brigham and Women's Hospital, Boston, MA, USA
[6]Department of Statistics and Actuarial Science, University of Hong Kong, Hong Kong, China
[7]Department of Biomedical Data Science, Radiology, and Medicine (Biomedical Informatics), Stanford University, Stanford, CA, USA
[8]Institute for Infocomm Research, Agency for Science Technology and Research (A*STAR), Singapore, Singapore
[9]Department of Radiology, Stanford University, Stanford, CA, USA
[10]University of Colorado School of Medicine, Aurora, CO, USA
[11]Department of Biostatistics and Health Data Science, Indiana University School of Medicine, Indianapolis, IN, USA
[12]Department of Radiology and Imaging Sciences, Indiana University School of Medicine, Indianapolis, IN, USA
[13]Department of Neurological Surgery, Indiana University School of Medicine, Indianapolis, IN, USA
[14]Department of Computer Science, Luddy School of Informatics, Computing, and Engineering, Indiana University, Indianapolis, IN, USA
[15]These authors contributed equally
[16]Senior authors
*Correspondence: spbakas@iu.edu

https://doi.org/10.1016/j.patter.2024.100974

THE BIGGER PICTURE Significant improvements can be made to clinical AI applications when multiple health-care institutions collaborate to build models that leverage large and diverse datasets. Federated learning (FL) provides a method for such AI model training, where each institution shares only model updates derived from their private training data, rather than the explicit patient data. This has been demonstrated to advance the state of the art for many clinical AI applications. However, open and persistent federations bring up the question of trust, and model updates have raised considerations of possible information leakage. Prior work has gone into understanding the inherent privacy risks and into developing mitigation techniques. Focusing on FL in health care, we review the privacy risks and the costs and limitations associated with state-of-the-art mitigations. We hope to provide a guide to health-care researchers seeking to engage in FL as a new paradigm of secure and private collaborative AI.

SUMMARY

Artificial intelligence (AI) shows potential to improve health care by leveraging data to build models that can inform clinical workflows. However, access to large quantities of diverse data is needed to develop robust generalizable models. Data sharing across institutions is not always feasible due to legal, security, and privacy concerns. Federated learning (FL) allows for multi-institutional training of AI models, obviating data sharing, albeit with different security and privacy concerns. Specifically, insights exchanged during FL can leak information about institutional data. In addition, FL can introduce issues when there is limited trust among the entities performing the compute. With the growing adoption of FL in health care, it is imperative to elucidate the potential risks. We thus summarize privacy-preserving FL literature in this work with special regard to health care. We draw attention to threats and review mitigation approaches. We anticipate this review to become a health-care researcher's guide to security and privacy in FL.

INTRODUCTION

The health-care domain has always dealt with privacy concerns and threats due to the sensitive nature of the information in the domain. For example, there have always been unauthorized access to medical records (which can be mitigated by requiring string access controls, user authentication, and audit logs),[1] insider threats that lead to misuse or inappropriate disclosure of health-care data (which can be mitigated by specific training, implementing policies that allow data access to employees where it is needed, and monitoring access to data),[2] and data breaches and/or cyber attacks (which can be mitigated by encrypting data, implementing robust network security policies, and regular security monitoring).[3] Although these challenges are critical, they have been documented and identified and are well studied. For the purposes of this review, we will be focusing on the use of advanced computational techniques in health care, where the privacy issues are more nuanced and their associated mitigation strategies are not that well studied compared to other fields.[4] The use of such tools (such as artificial intelligence [AI]) can make addressing these concerns more complicated due to the possibility that interactions with the application may leak information about the data used to train it.[3] Since health-care data are almost always tied with specific regulatory provisions,[5–7] privacy concerns of AI applications in this domain need a deeper understanding of the technical issues at hand, especially to provide guidelines for computational researchers developing algorithms and solutions in this field. This article aims to provide privacy and security guidelines for both computational researchers developing AI solutions in health care and regulatory authorities, so that they are mindful of both traditional information security concerns.

AI approaches have shown great promise for augmenting clinical workflows.[8] However, large and diverse datasets are required to train robust and generalizable AI models for clinical applications.[9–15] One method to acquire sufficient data is through multi-institutional collaborations, currently following a paradigm of centrally sharing data, also known as "data pooling"[16–20] (Figure 1A). However, such centralized data collection is not always possible due to various factors, such as patient privacy concerns, prohibitive costs of central data management and storage, and institutional or even regional data-sharing policies.[21,22] Federated learning (FL) is an alternative approach to the data pooling paradigm for multi-institutional collaborations that begins to address some privacy concerns, since model learning is performed locally at each institution and only the resulting local model parameters are shared (Figure 1B).

Although FL allows for training an AI model across private data without requiring that data to be shared, there are still questions that remain regarding the need and the way one needs to protect against leakage of information about these private data via the model updates shared throughout the FL workflow. There is hope that through the incorporation of additional security and privacy technologies into FL, a level of security and privacy can be achieved that will enable a greater degree of trust in the resulting federation. Many factors in addition to trust can prevent institutions from participating in FL training, such as coordination and overhead of data preparation, institutional information security, and compute hardware requirements. However, the use of more secure and private FL frameworks toward increasing trust in the system is expected to enable a more diverse collection of clinical institutions willing participate in FL projects. The models that result from such collaborations can potentially benefit from the data diversity, resulting in better model generalizability. In particular, secure and private FL has hope to greatly benefit collaborations in domains with stringent policies around data sharing, such as is the case for health care.[14,21–28] Some literature exists reporting on general vulnerabilities of FL[29,30] and even exploring privacy and security concepts related to FL,[31–33] albeit without providing a focus on their implications in the health-care domain. We further note a survey of open-source frameworks enabling FL, although without adequately exploring concepts of privacy.[34] Thus far, works that survey privacy issues related to FL in medicine[26,35–39] have provided details about FL in health care while not adequately expanding on the specific privacy threat associated with each attack.

In this work, we provide an overview of the current privacy threats and associated threat mitigations for FL workflows[40] while keeping the health-care context in mind. We summarize the key factors involved in determining the nature of privacy violation that can be related to each threat. Finally, we categorize the privacy threat mitigation technologies into distinct categories, and for each group we discuss what threats they address, as well as the costs associated with each mitigation technique. We hope that model developers having domain expertise can use this work to make more informed decisions with regard to patient privacy when using FL to train their models, and we hope to provide the context necessary for researchers to design experiments that most effectively advance our understanding of the problems and potential solutions for their domain.

In order to facilitate the focus on privacy concerns to FL deployments specifically in the health-care setting, we start by briefly introducing the concept of FL in health care. We then list the primary assets in AI that need to be protected in such an FL schema and then proceed with introducing a taxonomy of the potential threats to these assets, using the well-known confidentiality, integrity, and availability ("CIA") triad.[41] Notably, for each compromise, we consider the way in which it could adversely affect the collaborators. Finally, we discuss the mitigation methods that exist for these threats by considering how they reduce the impact of the threat, as well as the costs that are associated with the usage of the mitigation.

FEDERATED LEARNING

FL describes a collaborative approach to train AI models across decentralized collaborators (e.g., client servers on health-care sites) without directly sharing any training data between them.[22–25] This approach differs from the standard/traditional approach followed during training of AI models, which typically assumes that the data and model reside on a single, centralized location (see Figure 1 for an illustration). FL allows multiple institutions to train a single aggregate model without explicitly sharing any individual institution's patient data outside of that institution.

For example, to train a neural network (NN) with FL, an NN architecture needs to be chosen and code to implement training on this architecture incorporated into the system code to be distributed to all collaborators. We consider the use case of tumor boundary detection[14,22,24] for our explanation, where the trained

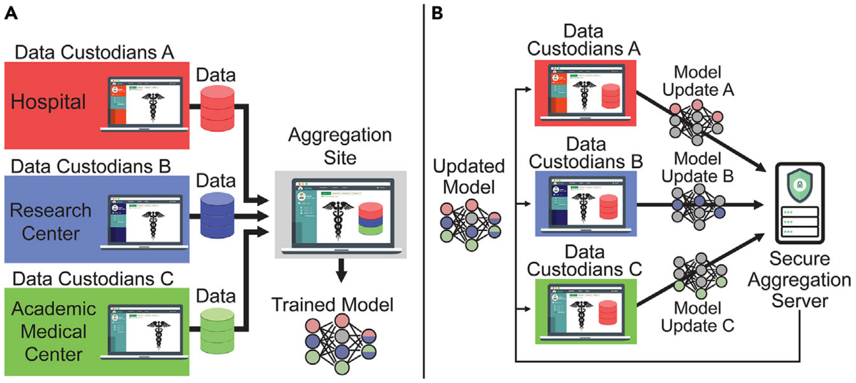

Figure 1. Illustration of different collaborative learning approaches

(A) Using a data-sharing paradigm, where data from the three individual data custodians are shared to a central data aggregation site for training and (B) using federated learning, where the training happens at each individual data custodian and only the model updates are shared to a secure aggregation server for combination.

AI model has an image as input and an image as output. The input image represents a clinically acquired scan, and the output is meant to identify regions of pathology associated with the presence of a cancerous tumor. The typical FL system consists of multiple participant sites, all independently connected via the network to a (central) "aggregator" node as depicted in Figure 1. To start the process, model initial weights are chosen at the aggregator as the initial global model and distributed via the network to all participants.

A typical "round" of federated training proceeds as follows. All participants perform training and validation on the aggregate model using their local data and compute resources and send their local model update as well as local validation scores to the aggregator node. The aggregator then aggregates all received local model updates and local validation scores, to form the updated aggregate model and aggregate validation scores for that round. This updated aggregate model is then sent back to all participants to initialize the next round of training. The complete course of FL training consists of multiple FL rounds, with a stopping/convergence condition and model selection criterion enforced by the aggregator using the model validation scores.

We take the opportunity to point out that model updates and validation scores should be viewed as a potential way to obtain information about the training input data. The local training described above consists of iteratively (1) passing batches of input images into the model, (2) measuring how well the model did at predicting the correct pathology locations, and (3) adjusting the model weights using small corrections obtained through the NN backpropagation process. Each batch of input data shifts the model weights using the information in that batch for performance to increase. This influence can potentially lead to information about the batches being detectable in the resulting model weights, as we will see in subsequent sections. The same is true for the validation metrics, although to a lesser degree. The intuition here is that patterns may be unique enough for one particular data sample that the results of training or validating on it could indicate its presence.

By being an approach in which data from one institute are never seen by another, it helps alleviate some privacy and security concerns, especially when dealing with sensitive medical data.[25,42] This potentially enables collaborations to be formed with larger and more diverse training data, which can result in trained models that generalize better.[22,43] In most FL algorithms, each institution independently performs training on their respective (local) data and provides the results of their computation (usually weights of a model) to be incorporated by the FL workflow (such as peer-to-peer aggregation or based on an aggregation server[25]; see Figure 1 for an illustration of FL using an aggregation server). Details of the typical central aggregator-based version that we provide as an example above can also differ depending on the implementation, for example, how model aggregation is performed or how the clients are selected in each round.[44–47]

FL has the potential to play a critical role in the next generation of privacy preservation strategies for health care, as evidenced by several recent high-impact studies.[14,22,24,25,27,48,49] The degree to which an FL system can be thought to be "secure and private" largely depends on the additional security and privacy technologies that it incorporates into the system. Concerns need to be addressed to mitigate malicious code execution, such as processes running at the collaborating institution's infrastructure that ex-filtrate their raw data or execute malicious forms of training. However, there are also threats from collaborators during FL that may attempt to extract information about the training data via the model updates that are shared between collaborators in the workflow. These updates may indeed leak information if in the hands of an adversary, as was suggested above.

SECURITY AND PRIVACY FOR FL IN HEALTH CARE

Importance of privacy in health care

Privacy attacks during the course of an FL collaboration result in exposure of data, model, or code (see Figure 2). Data privacy in health-care scenarios is crucial toward ensuring confidentiality and ethical handling of protected health information (PHI). Identifying robust mitigation techniques corresponding to specific threats, as well as implementing security measures, adopting encryption technologies, and adhering to privacy regulations (such as Health Insurance Portability and Accountability Act [HIPAA] from the United States[5] and General Data Protection Regulation [GDPR] from the European Union[6] or Digital Information Security in Healthcare Act [DISHA] from India[7]), is essential for safeguarding sensitive patient information and maintaining trust in health systems. Key health-care scenarios relevant to privacy preservation related to FL include (1) electronic health records (EHRs), which contain verbose textual data about a patient's medical history, procedures, diagnoses, medical scans, treatments, and medications, exposure of which would be a subject to patient confidentiality loss; (2) wearable devices, which collect health-related, fitness, and nutrition patient data directly associated with PHI; (3) local/institutional data repositories such as Biobanks and picture archiving and communication

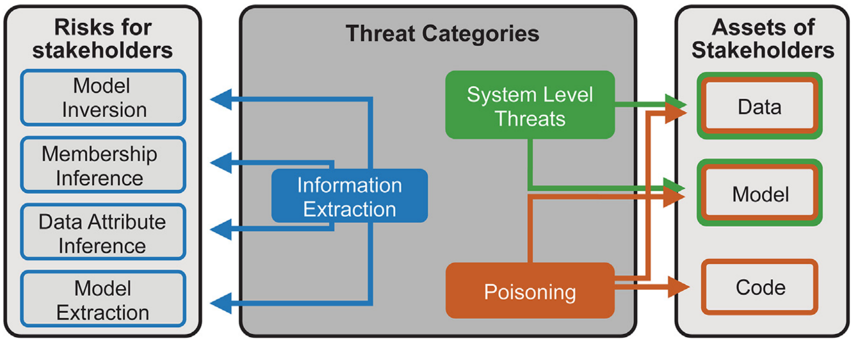

Figure 2. Illustration of overall privacy threat categories and their associated risks for stakeholders in a federated learning system
The "system-level threats" (in the green box) target the data and model; the "poisoning" attacks (in the orange box) attempt to expose the data, model, and code; and "information extraction" (in the blue box) targets model inversion, membership inference, data attribute inference, and model extraction.

systems (PACSs), including both data from routine clinical practice, clinical trials, and emergency services and unpublished research outcomes; (4) health insurance/billing; and (5) health-care policy.

Assets

There are three AI-specific assets to consider with regard to security and privacy around FL deployments. These are (1) the entire data cohort to be used for training; (2) the quantitative performance evaluation metrics that signify model performance, generated against both the local and the aggregated model updates; and (3) the model parameters themselves, for both the local and the aggregated model updates. In addition, the system on which the AI system is being deployed includes three additional assets that should be considered: (1) the hardware on which all the computations (i.e., training and FL aggregation) occur, (2) the actual source code for model training and FL execution, and (3) any additional metadata or configuration information that can be defined either in memory or as files on disk. Table 1 offers a summary of each asset and the properties we propose to address in this work. In the following sections, we will proceed through each of the CIA properties and elaborate on their meaning. On a high level, hardware protection is expected to already be in place, participants are expected to report correct validation metrics with what we see as minimal privacy consequences otherwise, and we see minimal privacy impact to participants dropping out and/or network connections being lost and so do not address issues of unavailable FL system resources.

Confidentiality

By "confidentiality" we refer to the degree to which the asset is hidden from others. As an example, if collaborator A sends their model update to the aggregator using transport layer security (TLS), then A has some assurances as to the confidentiality of the update while it is being transmitted to the aggregator. Once the aggregator receives the update and the decryption in TLS is performed, the degree to which the update remains confidential depends on what the aggregator code logic does with the update (for example, it could simply broadcast it to others), as well as how protected the aggregator processes are (e.g., code, memory, hardware instructions) from inspection by other processes and users on the aggregator infrastructure. These issues are exacerbated in the case of peer-to-peer aggregation methods,[25] where each collaborator performs weight aggregation on the model updates received by a peer-collaborator.

Table 1. Security and privacy assets in an FL system, including the CIA properties we propose to address in this work

Asset	Confidentiality	Integrity	Availability
Training data	✓	✓	✗
Quantitative metrics	✓	✗	✗
Model parameters	✓	✓	✗
Hardware	✗	✗	✗
Source code	✓	✓	✗
Additional files or information	✓	✓	✗

The confidentiality of any part of the data cohort and model parameters is considered here, with a break in confidentiality of either representing either a privacy violation or a leak of intellectual property (IP). Exposure of the complete data cohort in general can be a privacy violation, and model parameters can be used in an effort to reverse engineer training data.[50,51] Both model parameters and data can be considered IP, as both have value to organizations. We also consider exposure of approximations to these assets (i.e., data and model parameters) as a break in confidentiality. In the health-care setting, an approximation of a medical image may violate privacy just as much as the originally acquired image, and an approximation to model parameters may preserve enough model utility as to continue to hold a great deal of value as IP. The confidentiality of quantitative evaluation measures of model performance will also be considered, because such scores can be used to approximately recover the parameters of the model being evaluated, which, as discussed earlier, can further lead to approximate recovery of the underlying data.

Although physical isolation of hardware as well as the confidentiality of system code and additional files may be a general concern, they are considered out of the scope of this review, which instead focuses on the privacy and security aspects of medical data to be used in FL.

Integrity

By "integrity" we mean the degree to which the asset is precisely what it was expected to be. As an example, collaborator A may want to establish the integrity of the code running on the compute infrastructure of collaborator B. In some rare cases, though, A may trust B and their infrastructure to the extent that A is confident of such integrity.

The integrity of system hardware, i.e., being able to rely on the proper execution of hardware for a given hardware state, will be considered as out of scope of this work. This work shall instead focus on the integrity of training data and model parameters, as well as the integrity of system code and additional files and metadata. Although the integrity of model validation scores could be a concern in the general setting, we will not address this issue here, as the corruption of such scores would primarily be involved in an attack that was attempting to alter model selection (as such scores inform that process) and, as such, is not a significant concern in the health-care domain.

Availability

By ‘‘availability’’ we mean the degree to which an asset is available to use. As an example, a local model update may not be available at the aggregator if the network infrastructure at that collaborator is down, or the entire federation might get hampered if the network access at the aggregator level is lost.

Addressing availability issues for all listed assets in the previous section is considered out of the scope of this work, as we start with the assumption that the networking infrastructure for all collaborators in an FL system focusing on health care data is controlled by the respective clinical entities and, as such, is reliable and stable throughout the computation process.

THREATS TO PRIVACY DURING FL

In this section, we will be discussing the various threats to privacy in an FL system, an illustration of which can be found in Figure 2.

System-level threats

The threats in this category involve an adversary gaining access to either the raw data or the model weights. The adversary can acquire direct access through different means, such as privilege escalation and/or physical access, but always ends up with the requisite assets in their raw exact form. In contrast, all other threat categories in this section involve an adversary deriving approximations to these raw assets, such as extraction of training data information via the model weights[52] or manipulating their local data and/or training algorithm to exacerbate such an attack.

(1) Data ex-filtration: this involves an adversary obtaining access to the raw data (such as health-care scans or medical records) of one or multiple collaborators. The nature of privacy violation in this case is given by a patient’s right to not have their data given to anyone whom they did not explicitly authorize to have them.[5,6]

(2) Model ex-filtration: this involves an adversary obtaining access to the weights and biases of a local model update or global model aggregate(s). The primary concern would be either that the model represents IP or that the model could be used to extract information about the raw data used to train it.[52] Therefore, the nature of the privacy violation in this case can be the peculation of IP or could be any of the privacy violations[5,6] that can come from a successful attack to extract information about the training data from the model.

Information extraction

There are different types of privacy attack objectives related to the extraction of information about the training data from the model weights during, or at the conclusion of, the model training process. Multiple studies show that rare or unique parts of the training data are unintentionally retained by NNs.[52–54] The trained model weights transferred from a local institution (as well as the aggregate models they become a part of) can therefore be potentially exploited by any user with access to the model, taking advantage of this unintended memorization to gain sensitive information about the dataset being used by other collaborators in a federated setting. Some of the examples of such threats are illustrated below, and clinical researchers should consider the privacy impact of each of these threats independently.

After the attacks that use model access to approximate information about the data used to train it, we include one more attack that instead uses access to the model validation scores to approximate the model itself. This is an attack that is of concern in a federation that was otherwise acting to control access to the model (as IP) and, in addition, is a concern because such an approximate model can also be further used to approximate the training data themselves.

(1) Model inversion: such attacks involve an adversary with the ability to query the model or observe a model update, constructing a data sample meant to approximate an actual sample in a collaborator’s dataset.[55–57] Although the accuracy of these reconstructions can vary, the exposure of such a reconstruction may violate a patient’s privacy if features in the reconstruction are highly correlated to the original samples (for example, chest radiographs[48]). One form of this attack is carried out by an adversary with access to any version of the model and has been demonstrated outside of the FL context. Fredrikson et al.[56] showed how an adversary may use the prediction confidence values to approximate associated faces in a facial recognition system. Zhang et al.[58] demonstrated that an adversary with access to the model weights can approximate training examples for various classes using some auxiliary knowledge, such as blurred images from each of those classes. Other forms of this attack utilize single FL model updates from a particular collaborator and may make certain assumptions about the setting and attempt to approximate aspects of local training, such as batch normalization statistics or what labels were used for the batches.[59] State-of-the-art versions of these stronger attacks have demonstrated pixel-perfect reconstruction of images[60] when the attacker has access to local updates created with few samples, so that each sample has more relative influence. However, most FL round model updates are processed using many local data samples so that individual sample influence is reduced. In these cases, it is more difficult to reconstruct an exact training data sample from a local model update and even harder to reproduce one from the global aggregate model(s) it is included in. An overview of model inversion attack implementations and defense approaches is already described in prior work.[61] Advancements in these attacks continue to be put forward, and works that demonstrate such attacks in the setting of FL for medical models that demonstrate successful approximation of

hidden batch normalization statistics, for example,[62] acknowledge the importance of understanding such threats in these settings.

(2) Membership inference: membership inference is the process by which an adversary has possession of a particular data sample and is attempting to infer whether it was included in the training set of the model.[50,51,63] Exposure of whether a specific patient's data sample was used in training may be sensitive information, for example, if the presence of that sample implied something about the sample custodian (i.e., the dataset consisted of information about known felons or the dataset consisted of patients with a certain type of cancer). Success in accurately predicting which samples were involved in training a model is correlated with the degree to which a model encodes sample-specific information during training.[64,65] Due to this fact, membership inference attack success measurements are thought of as building blocks for state-of-the-art tools for generically determining the amount to which a model leaks information about its training data.[66] Clinicians and researchers should therefore consider successful membership inference attacks as a privacy concern, regardless of how compelling the concern is regarding the leakage of membership information alone. Such success may indicate that other more concerning attacks may be successful as well. For more details regarding the various membership inference attacks and defenses, see Hu et al.[67].

(3) Data attribute inference: instead of attempting to recover an entire data sample, this type of attack is characterized when an adversary attempts to recover only a subset of the data attributes from particular samples in the training set or, alternatively, the adversary attempts to learn attributes of the training set as a whole (such as aggregate statistical information). Such attacks have been demonstrated outside of the FL setting,[68] as well as within the FL setting with the attacker being an FL participant that only has access to the aggregated model,[50] although the use of alternative aggregation functions other than a simple weighted average could make this more difficult. Here, the attacker may estimate the accuracy of the reconstruction over other possible alternate values for the unknown data fields. Measures of confidence in the reconstruction may play a role in the impact of its exposure. Take the case of an attack to disclose the value ("true" or "false") of the attribute indicating a positive diagnosis of a particular medical condition. A confidence measure of 80% for such an attack, conditioned on a specific gender, location, and age of patient, may be used to assert that 80% of the samples in the training set corresponding to that specific gender, location, and age had a positive test result, which (depending on the characteristics) could be considered as exposure of PHI (defined as any information about health status, provision of health care, or payment for health care that is created or collected by a covered entity [or a business associate of a covered entity] and can be linked to a specific individual[5]) and, hence, identifying specific subjects from the training data. Such confidence measures have indeed been considered in previous studies and are easier to interpret when the feature space is relatively limited, such as when using categorical data with numerical digits (demonstrated by the attack example seeking to recover social security numbers[52]), and alternatives to standard model averaging, such as having collaborators withhold a subset of their local update, have been demonstrated to influence the effectiveness of such attacks.[50] Such confidence measurements would be more challenging to obtain for training sets containing only high-resolution images, for example.

(4) Model extraction: the traditional forms of this attack use the ability to obtain model outputs associated with arbitrary inputs chosen by the adversary[69,70] in order to estimate the model weights after a number of such queries. This is especially related to "black box"-style attacks, where a black box system[71] (basically, any system that does not expose any aspect of the inference process, starting from the data processing to the model weights, and only provides inference results) is used to generate outputs for multiple inputs in the form of predictions or validation scores. Depending on the amount of information provided by validation scores during FL training, such attacks based on validation scores alone may be more limited in their ability to approximate the model parameters, and with increasing numbers of input samples, the adversary can obtain a closer approximation to the model under attack. Such efforts to attain model weights, despite an appearance that access to such weights is restricted, could result in loss of valuable IP to a participant during a medical model federation.

Poisoning

This type of threat is specifically relevant for an adversary that is part of an FL system, such as a collaborator. A collaborator during the FL training process can maliciously alter their computations (by virtue of changing the local code, or data, or model) in order to magnify the effect of one of the attacks listed above on the private assets of others in the system, for example, by increasing data memorization. As such behavior serves to modify the model behavior in a malicious way, this falls under the category of model poisoning. Such advanced attacks are demonstrated for standard FL (where aggregation is simple model averaging) in Nasr et al.[51] and Hitaj et al.[72] and provide for a significant increase in the severity of the baseline attack. Alternative mechanisms for model averaging, such as limiting the portion of the local updates shared to the aggregator or employing a more robust aggregation, such as median rather than mean, can have an effect on the adversary's success but also affect the convergence properties of the global model. As such, these alternatives need to be considered together with the potential drawbacks to final model utility. Therefore, model poisoning can be considered fundamentally as a security attack by nature, since the primary attack vector is system asset corruption.

APPROACHES TO MITIGATE PRIVACY THREATS DURING FL

The amount of information that is exposed by the attacks described in the previous section can depend on the adversary's

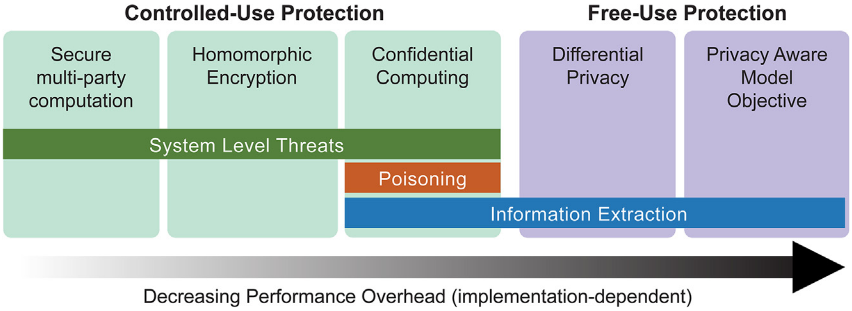
Controlled-Use Protection
Free-Use Protection
Secure multi-party computation
Homomorphic Encryption
Confidential Computing
Differential Privacy
Privacy Aware Model Objective
System Level Threats
Poisoning
Information Extraction
Decreasing Performance Overhead (implementation-dependent)

Figure 3. Various mitigation strategies and the respective privacy threats they aim to address
The "controlled-use protection" mechanisms of secure multi-party computation and homomorphic encryption are used to mitigate the system-level threats. The "free-use protection" mechanisms of differential privacy and privacy-aware model objective can be used to mitigate against information extraction. Confidential computing, on the other hand, can be used to counteract poisoning attacks in addition to the system-level threats and information extraction. The computational performance decreases as we move from secure multi-party computation to homomorphic encryption, confidential computing, differential privacy, and, finally, the privacy-aware model objective, being the most computationally inefficient.

level of access to the various assets in an FL system. As a basic requirement for considerations in health-care information technology (IT) infrastructures, we will assume that all FL systems will incorporate basic security measures, such as authentication to verify identities, support for encrypted communication, and other access control mechanisms meant to prevent exposure of assets to those who do not need to use them. However, since FL is a collaborative procedure, involving many parties who do need to handle assets in the system, the focus of the threat mitigations in this paper is to share assets and computational duties on those assets while mitigating the threats that exist when doing so with potentially untrusted parties. As an example, this includes the use of the final model, hosted by or with an untrusted user. As mentioned in the last section, such access can be sufficient for extracting model IP and/or carrying out membership or attribute inference attacks.

Some mitigations provide confidentiality of data while being used for computation, and others provide assurances that system code preapproved by participants matches identically to that being used at execution time. In addition, when an asset will be exposed due to requirements for its use, there are technologies to employ before the release of that asset to help mitigate a potential information extraction attack. We therefore consider two broad non-overlapping categories for the technologies that can be used during FL to mitigate the threats discussed in the previous section, which can be combined as needed (for an illustration, please refer to Figure 3): (1) controlled-use protection and (2) free-use protection technologies.

Controlled-use protection

These methods perform little to no alteration of the asset, but instead provide a way for potentially untrusted entities to perform computations on those assets without accessing the assets themselves, potentially controlling what computations are performed. None of the solutions in this category provide any protection against attempts to reverse engineer information from the outputs of the computations. For example, cryptography-based algorithms can be used to carry out remote algorithm execution with limitations on exposure of information to the remote parties during their calculation tasks (software-based confidentiality, such as homomorphic encryption [HE]).[73–75] In addition, specialized hardware solutions can provide computational resources while limiting the exposure of data used in the computation, by incorporating integrity checks during code execution to ensure that the appropriate code is being executed (hardware-based confidential computing [CC] with trusted execution).[76–78]

The first two solutions belonging to this category, i.e., secure multi-party computation (SMPC) and HE, allow for outsourcing computation with confidentiality of the inputs, intermediate results, and outputs and are implemented via software. These solutions may or may not provide assurances as to the integrity (correctness) of the computation. The other solution (i.e., confidential computing) is a hardware-based approach providing confidentiality of inputs, intermediate results, and outputs, while also providing assurances as to the integrity of the computation,[14] though usually with different assumptions as to in which circumstances protection is provided. CC solutions can generally help mitigate all threats listed in the previous section. Due to this broad threat coverage, combining CC with mitigation strategies that tackle broad information extraction threats (i.e., free-use protection category) can ensure the most robust security design. None of the solutions in this category provide any protection against reverse engineering of inputs to the computation itself. In general, there is a cost associated with these solutions that comes in the form of either increased computation and communication or special hardware requirements, in the case of CC.

Secure multi-party computation

SMPC[73–75] is an umbrella term that refers to a set of algorithms used to allow multiple entities to collectively calculate some function with controls as to what is exposed to the individual entities regarding both the inputs and the outputs of the function. As an example, suppose we wanted the aggregator in an aggregator-based FL[22,24,25] to know the output of a function of two inputs, and we want this to be computed using an input from each collaborator of a two-collaborator federation. The most basic example of an SMPC protocol here would be to use the trusted third party (TTP) protocol[79] to allow a third party (trusted by each collaborator and the aggregator) to take the inputs from each collaborator and send the output of the function applied to these inputs to the aggregator, without sharing either collaborator's inputs nor sharing the output to anyone except the aggregator. Using TTP, each collaborator would not learn anything new by participating in the protocol, and the aggregator only learns whatever can be deduced from the output of the function that was provided to it. Due to its simplicity, as well as the minimal information exposure involved, TTP is the benchmark protocol used to evaluate the properties of all other SMPC protocols.[79] Ongoing research is exploring the use of SMPC as a privacy enhancement to FL on medical data either by helping to prevent malicious models or by improving the confidentiality of model aggregation[80,81] or by making progress on the overhead that is incurred by its use.[82,83] These protocols can incur high computational and network communication overhead costs, as significant computation can be required to obfuscate information by encrypting/encoding and splitting into parts to avoid recovery, and significant communication protocols can be involved in order to coordinate the compute on the information pieces as well as combining the results to recover the function output without revealing information to unwanted parties in the process.

Homomorphic encryption

Although SMPC allows multiple institutions to jointly evaluate a function without needing to share their respective private inputs, the design of an SMPC protocol needs to take into account the specific function whose output is desired from the protocol, and a good deal of the protocol itself is dedicated to obfuscating the inputs. In contrast, HE[84–87] is a type of data encryption (and therefore provides cryptographic guarantees of confidentiality) that allows for generic computation on the data when in its encrypted form. The result of a computation on the encrypted data, when decrypted, is identical (or very close) to the result of the same computation performed on the unencrypted data. One benefit over SMPC is that multiple adversaries cannot collude to significantly increase the threat. However, the encryption for HE requires keys, and so key management is a necessary consideration here. HE by design provides a robust privacy solution for the application of FL.[88] However, almost all efforts in this regard suffer from a huge computational cost, and even an incremental increase in the data size (or in the NN layers) leads to an exponential increase in runtime. As such, more work needs to be done on improving the computational efficiency to render this approach practical for modern NN architectures. Although we list this as a software-based approach, efforts are ongoing to provide hardware acceleration for HE[89,90] (https://www.darpa.mil/news-events/2021-03-08), which has the potential to significantly reduce the overhead of this solution. It can be said that HE in its present technological development is most suited to those applications that are not time sensitive, where it can offer an extremely secure form of privacy preservation solution. A recent work by Froelicher et al.[91] shows the success of HE in providing truly private federated evaluation for applications within oncology and medical genetics, and work by Chen et al.[92] demonstrates success using HE for model aggregation during FL when transfer learning is used to reduce the size of the model weights that are processed using HE.

Confidential computing

In the previous points, we described methods that use encryption, encoding, or secret sharing to increase data confidentiality during computation. Alternatively, such confidentiality can also be obtained by means that are hardware enabled with so-called hardware-based CC with trusted execution,[78] though usually there are different assumptions here as to in which circumstances protection is provided. In CC, processes can be run inside so-called "enclaves" that essentially serve as a trusted third party in the sense that we used in the points related to SMPC and HE (not even privileged users on the system can access the memory or alter the execution). Code to run an algorithm can be put into the enclave, encrypted data can be passed in and unencrypted inside the enclave, then the algorithm can execute on the inputs and the result can be encrypted and passed out. In contrast to the previous solutions, these enclaves generally have the ability to attest to the fact that the code run inside the enclave was precisely what was expected, providing assurance as to the correctness of the result. Trust in the CC itself depends on trust in the hardware vendor that designs and distributes it, and trust in attestation will depend on trust in those who implement the service that carries it out. This feature allows for trust in all components of the FL system provided they are run with CC.[26] For example, during FL, running local training at a particular collaborator with CC can help prevent an insider adversary on that compute infrastructure from modifying their training code to compute updates using only a few data samples, in order to increase the ease with which another adversary could extract information about those few samples from the local updates sent from this collaborator. In addition, CC is more scalable and provides faster computation time compared to SMPC and HE.

Free-use protection

These approaches do nothing to protect data confidentiality while the computations are conducted and do nothing to ensure that the computations proceed as intended. However, they have the advantage of limiting how much of the result of the computation can be used to infer information about the original inputs. Therefore, solutions in this category are ideal for mitigating the threats of the previous section in the information extraction category. The costs associated with these solutions are increased computation and a reduction in the asset's utility due to its modification. For example, we will see that differentially private model training is a modification of a standard training algorithm that reduces the ability for someone with free access to the resulting model to carry out an information extraction attack that exposes information about the training data used to create it.

Differential privacy

When using the mitigation strategies described in the previous section for model training during FL, we are able to prevent data exposure during training. However, these techniques do nothing to prevent an adversary from using the trained model to reverse engineer information about (memorized/learned) training data, as is possible in a membership inference attack.[93] Differentially private model training, however, is a common approach toward mitigating the degree to which a model memorizes individual contributions to the data during training. It does so by introducing randomization during training to obfuscate the influence of these individual contributions while being able to learn over the data as a whole. DP algorithms come with privacy guarantees that relate to the likelihood that any single data point can be detected.[93]

An algorithm can be loosely defined as "differentially private" if the output of the algorithm cannot be used to distinguish whether a particular contribution to the data is present in the dataset used as input for the algorithm training.[93] Common examples of what type of contribution DP considers are those of a single data record or contributions of whole collaborator datasets. While the concepts surrounding DP were generally developed for use in data analytics, DP training algorithms have become a popular method for addressing user privacy concerns in AI.[94,95]

In the federated setting, DP algorithms can be used independently to train local model updates (local DP FL) or instead for the global consensus model aggregation (global DP FL). In local DP FL, each participating institution applies a DP training algorithm to perform their local training.[96] Here, the local model updates sent to the aggregator are produced with a DP privacy guarantee. This may be desired when the entity administering or running the aggregator is not trusted to prevent privacy attacks on its infrastructure. For global DP FL, the aggregation of model updates is made DP but there may be no guarantee with respect to the privacy of the local model updates handled by the aggregator.[97] If trust in the aggregator infrastructure is not already established, this may be done through the use of privacy solutions discussed in the previous section. Global DP FL is preferable to local, as it allows more data (all collaborator updates) to be combined before noising, which in principle improves the utility that is obtained for a given privacy level.[98]

Although DP has started gaining traction for deep-learning applications in medicine, it comes at the cost of a reduction in "model utility," which defines how well the model generalizes to new data when deployed,[99] as well as increased computation.[94] The model utility reduction comes from the noise addition during the training process, and the increased computation relates to potential changes in the way the training utilizes the underlying computational framework,[100] in addition to the potential need for more rounds of training. Importantly, DP training in FL could inhibit the use of data quality checks from specific collaborators, as privatized local model updates at the aggregator may mask signals that would otherwise indicate issues.

Survey articles for DP[101–104] can help to summarize the various approaches, best practices, and future research needed. However, more work is needed to understand the trade-offs associated with specific use cases, such as how much utility loss will be incurred at a given privacy level. We find in recent work[105–109] on DP FL training in medicine that federated training using DP (at $\epsilon = 4$, for example) can reach within 5% of the scores that would be achievable if DP were not used. As more research is done across different datasets and model architectures, a better understanding will form around how well these initial results will generalize. The privacy achieved by a DP algorithm should also be carefully considered. Most papers explore only ϵ values (lower indicates more privacy) greater than 1, and many explore ϵ values that are much greater. Since the worst-case odds of privacy exposure for an ϵ-DP algorithm is $e^{\epsilon} : 1$, the value $\epsilon = 4$, for example, is associated with worse than $50 : 1$ odds of privacy exposure.

Another complicating factor is the difficulty for data custodians to understand the privacy guarantee associated with the use of DP training, as it is very technical. In addition, the likelihood of specific privacy threats is even less likely to be understood until more research is done. This makes the proper balance of utility loss against true privacy concerns difficult to reason about, which is a very important aspect for the design of a practical solution.

Privacy-aware model objective

An alternative to making model training DP during FL is to incorporate the incentive against susceptibility to privacy attack into the training objective. Here an "attack model" is used during training in order to simulate a privacy attack on the primary model. Adversarial training is then performed at each collaborator during each round, alternating between improving a locally held attack model and improving the primary model by attempting to minimize a privacy-aware model objective (PAMO) (for example, the sum of the primary model loss and a measure of success for the attack model).[110,111] Such approaches may utilize mutual information estimations in order to establish a measure of success in minimizing information leakage or could use other measures to determine this success. The privacy protections afforded by this approach are similar to that of DP; however, the measure of protection is usually empirically based, in contrast to the theoretical privacy guarantee provided by DP. The costs associated with this approach are that of potential loss in model utility as well as potentially increased computation and overall local training required, as the local training is more complex.

An information conservation approach

In this section we describe a general approach that is meant to address the concern of information leakage from model updates concerning the individual data samples used during training. The mitigation measures discussed here are ones that restrict or obscure information during training, but without formal analysis on how such measures affect information flow. As such, an empirical assessment of the utility loss in the model as a result of each technique must be weighed against the ease of use, as well as any loss of privacy (calculated with privacy risk scores coming from empirical privacy vulnerability tools such as those discussed in Murakonda and Shokri[66] and Jayaraman and Evans[112]).

Some examples are as follows:

(1) Privacy-goal-oriented training methods: perform local training in a way that has been demonstrated as resistant to subsequent privacy attacks against the data used in training. In this approach, the training is tailored (e.g., by loss function or data pipeline specifications) to reduce memorization of the training data during training. Such efforts are described in Liu et al.[113] The PAMO mitigations of the previous section that are not accompanied by a measure of privacy afforded by the training fall into this category.

(2) Partial weight sharing: instead of sharing all the weights of the model to the aggregator during FL, only a predefined percentage (largest components) of the model is shared.[43,49]

DISCUSSION AND CONCLUSIONS

In this work, we provide a taxonomy and a deeper understanding of current privacy threats with their associated mitigation approaches, by keeping the focus on the context of FL in a health-care setting. We have provided common definitions that could be used in this field, while giving the reader detailed summaries of possible violations of privacy and their strategies to mitigate them, along with a meaningful categorization for both.

Table 2. Comparison of privacy-enhancing techniques in terms of properties that would need to be considered for deployment

Technique	SMPC	HE	CC	DP	PAMO
Exposure of data in use	no	no	no	yes	yes
Integrity of data in use	no	no	no	yes	no
Result unprotected from information extraction	yes	yes	yes	no	no
Execution integrity	depends on protocol	no	yes	no	no
Performance overhead (implementation dependent)	high	high	medium	medium	medium
Mathematical parity to the original results	yes	yes	yes	no	no
Threats mitigated	system threats	system threats	system threats, information extraction, poisoning	information extraction	information extraction

Each row is the property and the head of each column is the name of a privacy-enhancing technique.

We have begun to explore the veracity of these techniques in the context of FL for health care, following the mounting evidence that FL represents a potential paradigm shift on how multiple health-care institutions can collaborate to develop AI models without sharing any of their local data.[14,21,22,24,49,97,114] We hope that, by building upon previous works in the field,[26,29–32,34] we have provided an opportunity to current and future researchers in the field of health-care informatics to make better informed decisions during model training to appreciate potential security issues.

All the privacy threats and threat mitigation technique categories discussed in this review have been encapsulated in Table 2. The appropriate techniques to employ for a specific case differ due to the variety of protections afforded by each.

Although SMPC, HE, and CC (which protect the assets during controlled use) provide confidentiality of input data, as well as the intermediate results ("exposure of data in use"), they do not provide protection against an adversary reverse engineering this information from the final results ("results unprotected from information extraction"). In contrast, the DP and PAMO mitigation strategies (which provide free-use protection) alone do not address the confidentiality of input or intermediate results, but instead have the advantage of limiting the amount with which the final result of the computation can be used to infer information about the original inputs. Mitigations in the CC category may provide assurances as to the correctness of the computations being performed ("execution integrity"), whereas those in the other categories generally do not.

Although the costs associated with mitigation solutions can vary significantly, in general, the categories SMPC, HE, PAMO, and DP incur the cost of increased computation. In addition, SMPC can incur the cost of increased communication, and CC implementations in general have specific hardware requirements associated with enablement of hardware-supported trusted execution environments.[115] Finally, mitigations in the categories DP and PAMO generally incur the cost of a reduction in model utility (e.g., classification accuracy).

Protecting patient privacy must always be one of the primary considerations of health-care institutions, and it becomes more important as more clinical sites are initiating or joining collaborations that leverage health-care data to train AI models for further precision medicine.[14,16–20,116,117] As outlined in Kairouz et al.,[31] privacy attacks within the FL setting are a cause for concern. A significant amount of experimentation on the associated threats and mitigations, crucially in realistic scenarios on real-world data, is required to understand how they play out in different settings where health-care models are trained using FL. Understanding of the costs and benefits of these additional privacy protections during FL for health care is also critical, as there is mounting evidence[14,21,22,24,49,97,114] that FL with additional privacy protections may represent a potential paradigm shift on how multiple health-care institutions can collaborate to develop AI models without sharing any of their local data.

Application of the threat mitigation techniques outlined in this paper poses particular challenges in health care. There are no turnkey implementations, as solutions require careful configuration (and in some cases iterative tuning) to be effective. It is difficult to ensure that solutions will be accepted by the stakeholder involved, as policies on data security vary widely by institution. Even performing FL with no additional privacy threat mitigation can be difficult in the health-care domain, since most data that could be used for these purposes reside in private institutions that lack incentives to participate in FL activities, and often institutions use operational systems for data handling that make connecting the data to FL platforms a difficult task. This is in addition to the requirement that institutions export the data cohort and de-identify them prior to starting any research. These issues, specific to the health-care setting, will need to be addressed as research activities in privacy-threat-mitigating machine learning (ML) model training for medicine are undertaken. Solutions for these issues require community-driven standards to be developed in concert with relevant stakeholders.[118] Otherwise, the legal risks of loss of patient privacy due to an improperly designed solution may outweigh the benefits of integrating AI models in clinical workflows.

We are still in the early days of considering these privacy concerns and using the security technologies highlighted in this review in large-scale FL deployments. There is a significant amount of research to be done, especially into how the incorporation of select privacy mitigations into a federated study will impact the stakeholders involved. As we discussed above, the costs and benefits of individual solutions are being explored and improved in the literature against standard measures. The results are frequently dependent on specifics such as the trained algorithm, the data distributions involved, and the compute and physical network infrastructure to be used. More studies are needed to get a better understanding of how the current results will generalize in large-scale federations. In addition, it can be difficult for prospective FL participants to make decisions based only on the measures of privacy benefit currently reported for a solution in the literature. These measures may not map well to either regulation or common patient privacy concerns. Although some work exists, more research to help bridge this gap would be valuable for those trying to balance the concerns of stakeholders to a federation.[39,45,61,67,82,104,119]

In conclusion, this review encapsulates and illustrates some of the major research directions pertaining to privacy in FL, and we hope it can be used as a primer and reference for future research studies as security becomes a growing concern in the healthcare informatics community. Although a lot of work has been done in this area, more detailed experimentation of these methods in realistic scenarios with ample, diverse, and clinically relevant use cases will be essential for their proper quantification and subsequent evaluation for clinical deployment.

ACKNOWLEDGMENTS

Research reported in this publication was partly supported by the National Cancer Institute (NCI) of the National Institutes of Health (NIH) under award nos. U01CA242871, U24CA279629, and U01CA242879. The content of this publication is solely the responsibility of the authors and does not represent the official views of the NIH.

S.P. and S.B. conducted part of the work reported in this article at their current affiliation, as well as while they were affiliated with the Center for Artificial Intelligence and Data Science for Integrated Diagnostics (AI2D), the Center for Biomedical Image Computing and Analytics (CBICA), and the Departments of Radiology and of Pathology and Laboratory Medicine, Perelman School of Medicine at the University of Pennsylvania, Philadelphia, Pennsylvania 19104, USA. Fredrik Hans Skarstedt (Indiana University) helped create the figures.

AUTHOR CONTRIBUTIONS

All authors contributed to the writing and editing of the manuscript.

DECLARATION OF INTERESTS

The authors declare no competing interests.

REFERENCES

1. Moore, W., and Frye, S. (2019). Review of hipaa, part 1: history, protected health information, and privacy and security rules. J. Nucl. Med. Technol. *47*, 269–272. https://doi.org/10.2967/jnmt.119.227892.

2. Mercuri, R.T. (2004). The hipaa-potamus in health care data security. Commun. ACM *47*, 25–28.

3. Choi, Y.B., Capitan, K.E., Krause, J.S., and Streeper, M.M. (2006). Challenges associated with privacy in health care industry: implementation of hipaa and the security rules. J. Med. Syst. *30*, 57–64. https://doi.org/10.1007/s10916-006-7405-0.

4. Usynin, D., Rueckert, D., Passerat-Palmbach, J., and Kaissis, G. (2022). Zen and the art of model adaptation: Low-utility-cost attack mitigations in collaborative machine learning. Proc. Priv. Enhanc. Technol. *2022*, 274–290. https://doi.org/10.2478/popets-2022-0014.

5. Annas, G.J., et al. (2003). Hipaa regulations-a new era of medical-record privacy? N. Engl. J. Med. *348*, 1486–1490. https://doi.org/10.1056/NEJMlim035027.

6. Voigt, P., and Von dem Bussche, A. (2017). The eu general data protection regulation (gdpr). In A Practical Guide, 1st Ed., *10* (Springer International Publishing), pp. 3152676. https://doi.org/10.5555/3152676.

7. Haidar, M., and Kumar, S. (2021). Smart healthcare system for biomedical and health care applications using aadhaar and blockchain. In 2021 5th International Conference on Information Systems and Computer Networks (ISCON) (IEEE), pp. 1–5. https://doi.org/10.1109/ISCON52037.2021.9702306.

8. Topol, E.J. (2019). High-performance medicine: the convergence of human and artificial intelligence. Nat. Med. *25*, 44–56. https://doi.org/10.1038/s41591-018-0300-7.

9. Dunnmon, J.A., Yi, D., Langlotz, C.P., Ré, C., Rubin, D.L., and Lungren, M.P. (2019). Assessment of convolutional neural networks for automated classification of chest radiographs. Radiology *290*, 537–544. https://doi.org/10.1148/radiol.2018181422.

10. AlBadawy, E.A., Saha, A., and Mazurowski, M.A. (2018). Deep learning for segmentation of brain tumors: Impact of cross-institutional training and testing. Med. Phys. *45*, 1150–1158. https://doi.org/10.1002/mp.12752.

11. Chang, K., Beers, A.L., Brink, L., Patel, J.B., Singh, P., Arun, N.T., Hoebel, K.V., Gaw, N., Shah, M., Pisano, E.D., et al. (2020). Multi-institutional assessment and crowdsourcing evaluation of deep learning for automated classification of breast density. J. Am. Coll. Radiol. *17*, 1653–1662. https://doi.org/10.1016/j.jacr.2020.05.015.

12. Pati, S., Thakur, S.P., Bhalerao, M., Thermos, S., Baid, U., Gotkowski, K., Gonzalez, C., Guley, O., Hamamci, I.E., Er, S., et al. (2021a). Gandlf: A generally nuanced deep learning framework for scalable end-to-end clinical workflows in medical imaging. Preprint at arXiv. https://doi.org/10.48550/arXiv.2103.01006.

13. Thakur, S.P., Schindler, M.K., Bilello, M., and Bakas, S. (2022). Clinically deployed computational assessment of multiple sclerosis lesions. Front. Med. *9*, 797586. https://doi.org/10.3389/fmed.2022.797586.

14. Pati, S., Baid, U., Edwards, B., Sheller, M., Wang, S.-H., Reina, G.A., Foley, P., Gruzdev, A., Karkada, D., Davatzikos, C., et al. (2022). Federated learning enables big data for rare cancer boundary detection. Nat. Commun. *13*, 7346. https://doi.org/10.1038/s41467-022-33407-5.

15. Pati, S., Thakur, S.P., Hamamcı, İ.E., Baid, U., Baheti, B., Bhalerao, M., Güley, O., Mouchtaris, S., Lang, D., Thermos, S., et al. (2023). Gandlf: the generally nuanced deep learning framework for scalable end-to-end clinical workflows. Commun. Eng. *2*, 23. https://doi.org/10.1038/s44172-023-00066-3.

16. GLASS Consortium (2018). Glioma through the looking glass: molecular evolution of diffuse gliomas and the glioma longitudinal analysis consortium. Neuro Oncol. *20*, 873–884. https://doi.org/10.1093/neuonc/noy020.

17. Bakas, S., Ormond, D.R., Alfaro-Munoz, K.D., Smits, M., Cooper, L.A.D., Verhaak, R., and Poisson, L.M. (2020). iglass: imaging integration into the glioma longitudinal analysis consortium. Neuro Oncol. *22*, 1545–1546. https://doi.org/10.1093/neuonc/noaa160.

18. Davatzikos, C., Barnholtz-Sloan, J.S., Bakas, S., Colen, R., Mahajan, A., Quintero, C.B., Capellades Font, J., Puig, J., Jain, R., Sloan, A.E., et al. (2020). Ai-based prognostic imaging biomarkers for precision neuro-oncology: the respond consortium. Neuro Oncol. *22*, 886–888. https://doi.org/10.1093/neuonc/noaa045.

19. Bakas, S., Reyes, M., Jakab, A., Bauer, S., Rempfler, M., Crimi, A., Shinohara, R.T., Berger, C., Ha, S.M., Rozycki, M., et al. (2018). Identifying the best machine learning algorithms for brain tumor segmentation, progression assessment, and overall survival prediction in the brats challenge. Preprint at arXiv. https://doi.org/10.48550/arXiv.1811.02629.

20. Baid, U., Ghodasara, S., Bilello, M., Mohan, S., Calabrese, E., Colak, E., Farahani, K., Kalpathy-Cramer, J., Kitamura, F.C., Pati, S., et al. (2021). The rsna-asnr-miccai brats 2021 benchmark on brain tumor segmentation and radiogenomic classification. Preprint at arXiv. https://doi.org/10.48550/arXiv.2107.02314.

21. Chang, K., Balachandar, N., Lam, C., Yi, D., Brown, J., Beers, A., Rosen, B., Rubin, D.L., and Kalpathy-Cramer, J. (2018). Distributed deep learning networks among institutions for medical imaging. J. Am. Med. Inf. Assoc. *25*, 945–954. https://doi.org/10.1093/jamia/ocy017.

22. Sheller, M.J., Edwards, B., Reina, G.A., Martin, J., Pati, S., Kotrotsou, A., Milchenko, M., Xu, W., Marcus, D., Colen, R.R., and Bakas, S. (2020). Federated learning in medicine: facilitating multi-institutional collaborations without sharing patient data. Sci. Rep. *10*, 12598–12612. https://doi.org/10.1038/s41598-020-69250-1.

23. McMahan, H.B., Moore, E., Ramage, D., Hampson, S., et al. (2016). Communication-efficient learning of deep networks from decentralized data. Preprint at arXiv. https://doi.org/10.48550/arXiv.1602.05629.

24. Sheller, M.J., Reina, G.A., Edwards, B., Martin, J., and Bakas, S. (2018). Multi-institutional deep learning modeling without sharing patient data: A feasibility study on brain tumor segmentation. In International MICCAI Brainlesion Workshop (Springer), pp. 92–104. https://doi.org/10.1007/978-3-030-11723-8_9.

25. Rieke, N., Hancox, J., Li, W., Milletarì, F., Roth, H.R., Albarqouni, S., Bakas, S., Galtier, M.N., Landman, B.A., Maier-Hein, K., et al. (2020). The future of digital health with federated learning. NPJ Digit. Med. *3*, 119–127. https://doi.org/10.1038/s41746-020-00323-1.

26. Kaissis, G.A., Makowski, M.R., Rückert, D., and Braren, R.F. (2020). Secure, privacy-preserving and federated machine learning in medical imaging. Nat. Mach. Intell. *2*, 305–311. https://doi.org/10.1038/s42256-020-0186-1.

27. Roth, H.R., Chang, K., Singh, P., Neumark, N., Li, W., Gupta, V., Gupta, S., Qu, L., Ihsani, A., Bizzo, B.C., et al. (2020). Federated learning for breast density classification: A real-world implementation. In Domain Adaptation and Representation Transfer, and Distributed and Collaborative Learning (Springer), pp. 181–191. https://doi.org/10.1007/978-3-030-60548-3_18.

28. Qu, L., Balachandar, N., Zhang, M., and Rubin, D. (2022). Handling data heterogeneity with generative replay in collaborative learning for medical imaging. Med. Image Anal. *78*, 102424. https://doi.org/10.1016/j.media.2022.102424.

29. Mothukuri, V., Parizi, R.M., Pouriyeh, S., Huang, Y., Dehghantanha, A., and Srivastava, G. (2021). A survey on security and privacy of federated learning. Future Generat. Comput. Syst. *115*, 619–640. https://doi.org/10.1016/j.future.2020.10.007.

30. Bouacida, N., and Mohapatra, P. (2021). Vulnerabilities in federated learning. IEEE Access *9*, 63229–63249. https://doi.org/10.1109/ACCESS.2021.3075203.

31. Kairouz, P., McMahan, H.B., Avent, B., Bellet, A., Bennis, M., Nitin Bhagoji, A., Bonawitz, K., Charles, Z., Cormode, G., Cummings, R., et al. (2021). Advances and open problems in federated learning. FNT. in Machine Learning *14*, 1–210. https://doi.org/10.1561/2200000083.

32. Li, T., Sahu, A.K., Talwalkar, A., and Smith, V. (2020). Federated learning: Challenges, methods, and future directions. IEEE Signal Process. Mag. *37*, 50–60. https://doi.org/10.1109/MSP.2020.2975749.

33. Hatamizadeh, A., Yin, H., Molchanov, P., Myronenko, A., Li, W., Dogra, P., Feng, A., Flores, M.G., Kautz, J., Xu, D., et al. (2022a). Do gradient inversion attacks make federated learning unsafe?. Preprint at arXiv. https://doi.org/10.48550/arXiv.2202.06924.

34. Li, Q., Wen, Z., Wu, Z., Hu, S., Wang, N., Li, Y., Liu, X., and He, B. (2023). A survey on federated learning systems: vision, hype and reality for data privacy and protection. IEEE Trans. Knowl. Data Eng. *35*, 3347–3366. https://doi.org/10.1109/TKDE.2021.3124599.

35. Aouedi, O., Sacco, A., Piamrat, K., and Marchetto, G. (2023). Handling privacy-sensitive medical data with federated learning: Challenges and future directions. IEEE J. Biomed. Health Inform. *27*, 790–803. https://doi.org/10.1109/JBHI.2022.3185673.

36. Xu, J., Glicksberg, B.S., Su, C., Walker, P., Bian, J., and Wang, F. (2021). Federated learning for healthcare informatics. J. Healthc. Inform. Res. *5*, 1–19. https://doi.org/10.1007/s41666-020-00082-4.

37. Prayitno Shyu, C.-R., Shyu, C.R., Putra, K.T., Chen, H.C., Tsai, Y.Y., Hossain, K.S.M.T., Jiang, W., and Shae, Z.Y. (2021a). A systematic review of federated learning in the healthcare area: From the perspective of data properties and applications. Appl. Sci. *11*, 11191. https://doi.org/10.3390/app112311191.

38. Antunes, R.S., André da Costa, C., Küderle, A., Yari, I.A., and Eskofier, B. (2022). Federated learning for healthcare: Systematic review and architecture proposal. ACM Trans. Intell. Syst. Technol. *13*, 1–23. https://doi.org/10.1145/3501813.

39. de Castro, L., Agrawal, R., Yazicigil, R., Chandrakasan, A., Vaikuntanathan, V., Juvekar, C., and Joshi, A. (2021). Does fully homomorphic encryption need compute acceleration?. Preprint at arXiv. https://doi.org/10.48550/arXiv.2112.06396.

40. Vassilev, A., Oprea, A., Fordyce, A., and Anderson, H. (2024). Adversarial machine learning: A taxonomy and terminology of attacks and mitigations. Tech. Rep. National Institute of Standards and Technology. https://doi.org/10.6028/NIST.AI.100-2e2023.

41. Ham, J.V.D. (2021). Toward a better understanding of “cybersecurity”. Digital Threats. *2*, 1–3. https://doi.org/10.1145/3442445.

42. Prayitno Shyu, C.-R., Shyu, C.R., Putra, K.T., Chen, H.C., Tsai, Y.Y., Hossain, K.S.M.T., Jiang, W., and Shae, Z.Y. (2021b). A systematic review of federated learning in the healthcare area: From the perspective of data properties and applications. Appl. Sci. *11*, 11191. https://doi.org/10.3390/app112311191.

43. Li, W., Milletarì, F., Xu, D., Rieke, N., Hancox, J., Zhu, W., Baust, M., Cheng, Y., Ourselin, S., Cardoso, M.J., et al. (2019). Privacy-preserving federated brain tumour segmentation. In International workshop on machine learning in medical imaging (Springer), pp. 133–141. https://doi.org/10.1007/978-3-030-32692-0_16.

44. Qi, P., Chiaro, D., Guzzo, A., Ianni, M., Fortino, G., and Piccialli, F. (2024). Model aggregation techniques in federated learning: A comprehensive survey. Future Generat. Comput. Syst. *150*, 272–293. https://doi.org/10.1016/j.future.2023.09.008.

45. Zhang, G., Liu, B., Zhu, T., Zhou, A., and Zhou, W. (2022). Visual privacy attacks and defenses in deep learning: a survey. Artif. Intell. Rev. *55*, 4347–4401. https://doi.org/10.1007/s10462-021-10123-y.

46. Smestad, C., and Li, J. (2023). A systematic literature review on client selection in federated learning. In Proceedings of the 27th International Conference on Evaluation and Assessment in Software Engineering, pp. 2–11. https://doi.org/10.1145/3593434.3593438.

47. Huang, J., Hong, C., Liu, Y., Chen, L.Y., and Roos, S. (2023). Maverick matters: Client contribution and selection in federated learning. In Pacific-Asia Conference on Knowledge Discovery and Data Mining (Springer), pp. 269–282. https://doi.org/10.1007/978-3-031-33377-4_21.

48. Kaissis, G., Ziller, A., Passerat-Palmbach, J., Ryffel, T., Usynin, D., Trask, A., Lima, I., Mancuso, J., Jungmann, F., Steinborn, M.-M., et al. (2021). End-to-end privacy preserving deep learning on multi-institutional medical imaging. Nat. Mach. Intell. *3*, 473–484. https://doi.org/10.1038/s42256-021-00337-8.

49. Dayan, I., Roth, H.R., Zhong, A., Harouni, A., Gentili, A., Abidin, A.Z., Liu, A., Costa, A.B., Wood, B.J., Tsai, C.-S., et al. (2021). Federated learning for predicting clinical outcomes in patients with covid-19. Nat. Med. *27*, 1735–1743. https://doi.org/10.1038/s41591-021-01506-3.

50. Melis, L., Song, C., De Cristofaro, E., and Shmatikov, V. (2019). Exploiting unintended feature leakage in collaborative learning. Preprint at arXiv. https://doi.org/10.1109/SP.2019.00029.

51. Nasr, M., Shokri, R., and Houmansadr, A. (2019). Comprehensive privacy analysis of deep learning: Passive and active white-box inference attacks against centralized and federated learning. In 2019 IEEE symposium on security and privacy (SP) (IEEE), pp. 739–753. https://doi.org/10.1109/SP.2019.00065.

52. Carlini, N., Liu, C., Kos, J., Erlingsson, Ú., and Song, D. (2018). The secret sharer: Measuring unintended neural network memorization & extracting secrets. Preprint at arXiv. https://doi.org/10.48550/arXiv.1802.08232.

53. Thakkar, O.D., Ramaswamy, S., Mathews, R., and Beaufays, F. (2021). Understanding unintended memorization in language models under federated learning. In Proceedings of the Third Workshop on Privacy in

Natural Language Processing, pp. 1–10. https://doi.org/10.18653/v1/2021.privatenlp-1.1.

54. Song, C., Ristenpart, T., and Shmatikov, V. (2017). Machine learning models that remember too much. In Proceedings of the 2017 ACM SIGSAC Conference on computer and communications security, pp. 587–601. https://doi.org/10.1145/3133956.3134077.

55. Fredrikson, M., Lantz, E., Jha, S., Lin, S., Page, D., and Ristenpart, T. (2014). Privacy in pharmacogenetics: An end-to-end case study of personalized warfarin dosing. In Proceedings of the 23rd USENIX Conference on Security Symposium. SEC'14 USA (USENIX Association), pp. 17–32. https://doi.org/10.5555/2671225.2671227.

56. Fredrikson, M., Jha, S., and Ristenpart, T. (2015). Model inversion attacks that exploit confidence information and basic countermeasures. In Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security. CCS '15 (Association for Computing Machinery), pp. 1322–1333. URL:. https://doi.org/10.1145/2810103.2813677

57. Li, Z., Wang, L., Chen, G., Zhang, Z., Shafiq, M., and Gu, Z. (2023). E2egi: End-to-end gradient inversion in federated learning. IEEE J. Biomed. Health Inform. *27*, 756–767. https://doi.org/10.1109/JBHI.2022.3204455.

58. Zhang, Y., Jia, R., Pei, H., Wang, W., Li, B., and Song, D. (2020). The secret revealer: Generative model-inversion attacks against deep neural networks. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 253–261.

59. Huang, Y., Gupta, S., Song, Z., Li, K., and Arora, S. (2021). Evaluating gradient inversion attacks and defenses in federated learning. Adv. Neural Inf. Process. Syst. *34*, 7232–7241.

60. Zhu, L., Liu, Z., and Han, S. (2019). Deep leakage from gradients. Adv. Neural Inf. Process. Syst. *32*.

61. Song, J., and Namiot, D. (2022). A survey of the implementations of model inversion attacks. In International Conference on Distributed Computer and Communication Networks (Springer), pp. 3–16. https://doi.org/10.1007/978-3-031-30648-8_1.

62. Hatamizadeh, A., Yin, H., Molchanov, P., Myronenko, A., Li, W., Dogra, P., Feng, A., et al. (2022b). Do gradient inversion attacks make federated learning unsafe?. Preprint at arXiv. https://doi.org/10.48550/arXiv.2202.06924.

63. Liu, Y., Wen, R., He, X., Salem, A., Zhang, Z., Backes, M., De Cristofaro, E., Fritz, M., and Zhang, Y. (2022a). {ML-Doctor}: Holistic risk assessment of inference attacks against machine learning models. In 31st USENIX Security Symposium (USENIX Security 22), pp. 4525–4542.

64. Samala, R.K., Chan, H.-P., Hadjiiski, L., and Koneru, S. (2020). Hazards of data leakage in machine learning: a study on classification of breast cancer using deep neural networks. Medical Imaging 2020: Computer-Aided Diagnosis *11314*, 279–284. https://doi.org/10.1117/12.2549313.

65. Kaufman, S., Rosset, S., Perlich, C., and Stitelman, O. (2012). Leakage in data mining: Formulation, detection, and avoidance. ACM Trans. Knowl. Discov. Data *6*, 1–21. https://doi.org/10.1145/2382577.2382579.

66. Murakonda, S.K., and Shokri, R. (2020). Ml privacy meter: Aiding regulatory compliance by quantifying the privacy risks of machine learning. Preprint at arXiv. missingarXiv:2007.09339. https://doi.org/10.48550/arXiv.2007.09339.

67. Hu, H., Salcic, Z., Sun, L., Dobbie, G., Yu, P.S., and Zhang, X. (2022). Membership inference attacks on machine learning: A survey. ACM Comput. Surv. *54*, 1–37. https://doi.org/10.1145/3523273.

68. Ateniese, G., Mancini, L.V., Spognardi, A., Villani, A., Vitali, D., and Felici, G. (2015). Hacking smart machines with smarter ones: How to extract meaningful data from machine learning classifiers. Int. J. Secur. Network. *10*, 137–150. https://doi.org/10.1504/IJSN.2015.071829.

69. Sanyal, S., Addepalli, S., and Babu, R.V. (2022). Towards data-free model stealing in a hard label setting. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 15284–15293. https://doi.org/10.1109/CVPR52688.2022.01485.

70. Orekondy, T., Schiele, B., and Fritz, M. (2019). Knockoff nets: Stealing functionality of black-box models. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 4954–4963. https://doi.org/10.1109/CVPR.2019.00509.

71. Guidotti, R., Monreale, A., Ruggieri, S., Turini, F., Giannotti, F., and Pedreschi, D. (2018). A survey of methods for explaining black box models. ACM Comput. Surv. *51*, 1–42. https://doi.org/10.1145/3236009.

72. Hitaj, B., Ateniese, G., and Perez-Cruz, F. (2017). Deep Models under the gan: Information Leakage from Collaborative Deep Learning. https://doi.org/10.1145/3133956.3134012.

73. Yao, A.C. (1982). Protocols for secure computations. In 23rd annual symposium on foundations of computer science (sfcs 1982) (IEEE), pp. 160–164. https://doi.org/10.1109/SFCS.1982.38.

74. Goldreich, O. (1998). Secure multi-party computation. Manuscript. Preliminary version *78*, 110.

75. Shamir, A., Rivest, R.L., and Adleman, L.M. (1981). Mental poker. In The mathematical gardner (37–43) (Springer), pp. 37–43. https://doi.org/10.1007/978-1-4684-6686-7_5.

76. Sabt, M., Achemlal, M., and Bouabdallah, A. (2015). Trusted execution environment: what it is, and what it is not. In 2015 IEEE Trustcom/BigDataSE/ISPA, *1* (IEEE), pp. 57–64. https://doi.org/10.1109/Trustcom.2015.357.

77. Schneider, M., Masti, R.J., Shinde, S., Capkun, S., and Perez, R. (2022). Sok: Hardware-supported trusted execution environments. Preprint at arXiv. https://doi.org/10.48550/arXiv.2205.12742.

78. Consortium, C., et al. (2021). Confidential computing: Hardware-based trusted execution for applications and data. A Publication of The Confidential Computing Consortium.

79. Frikken, K.B. (2010). Secure multiparty computation. Algorithms and theory of computation handbook: special topics and techniques (ACM), p. 14. https://doi.org/10.5555/1882723.1882737.

80. Kalapaaking, A.P., Stephanie, V., Khalil, I., Atiquzzaman, M., Yi, X., and Almashor, M. (2022). Smpc-based federated learning for 6g-enabled internet of medical things. IEEE Network *36*, 182–189. https://doi.org/10.1109/MNET.007.2100717.

81. Kalapaaking, A.P., Khalil, I., and Yi, X. (2023). Blockchain-based federated learning with smpc model verification against poisoning attack for healthcare systems. Preprint at arXiv. https://doi.org/10.48550/arXiv.2304.13360.

82. Buyukates, B., So, J., Mahdavifar, H., and Avestimehr, S. (2022). Lightverifl: Lightweight and verifiable secure federated learning. In Workshop on Federated Learning: Recent Advances and New Challenges (in Conjunction with NeurIPS 2022), pp. 1–20. URL: https://openreview.net/pdf?id=WA7I-Fm4tmP

83. Huang, C., Yao, Y., Zhang, X., Teng, D., Wang, Y., and Zhou, L. (2022). Robust secure aggregation with lightweight verification for federated learning. In 2022 IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom) (IEEE), pp. 582–589. https://doi.org/10.1109/TrustCom56396.2022.00085.

84. Gentry, C., and Halevi, S. (2011). Implementing gentry's fully-homomorphic encryption scheme. In Annual international conference on the theory and applications of cryptographic techniques (Springer), pp. 129–148. https://doi.org/10.1007/978-3-642-20465-4_9.

85. Ahmed, E.-Y., and ELKETTANI, M.D. (2016). Fully homomorphic encryption: state of art and comparison. Int. J. Comput. Sci. Inf. Secur. *14*. https://doi.org/10.6084/M9.FIGSHARE.3362338.

86. Acar, A., Aksu, H., Uluagac, A.S., and Conti, M. (2018). A survey on homomorphic encryption schemes: Theory and implementation. ACM Comput. Surv. *51*, 1–35. https://doi.org/10.1145/3214303.

87. Stripelis, D., Saleem, H., Ghai, T., Dhinagar, N.J., Gupta, U., Anastasiou, C., Ver Steeg, G., Ravi, S., Naveed, M., Thompson, P.M., and Ambite, J.L. (2021). Secure neuroimaging analysis using federated learning with homomorphic encryption. Preprint at arXiv. https://doi.org/10.1117/12.2606256.

88. Ma, J., Naas, S.-A., Sigg, S., and Lyu, X. (2021). Privacy-preserving federated learning based on multi-key homomorphic encryption. Preprint at arXiv. https://doi.org/10.1002/int.22818.

89. Doröz, Y., Öztürk, E., and Sunar, B. (2014). Accelerating fully homomorphic encryption in hardware. IEEE Trans. Comput. *64*, 1–1521. https://doi.org/10.1109/TC.2014.2345388.

90. Cao, X., Moore, C., O'Neill, M., O'Sullivan, E., and Hanley, N. (2013). Accelerating fully homomorphic encryption over the integers with super-size hardware multiplier and modular reduction. Cryptology ePrint Archive. URL: https://eprint.iacr.org/2013/616

91. Froelicher, D., Troncoso-Pastoriza, J.R., Raisaro, J.L., Cuendet, M.A., Sousa, J.S., Cho, H., Berger, B., Fellay, J., and Hubaux, J.-P. (2021). Truly privacy-preserving federated analytics for precision medicine with multiparty homomorphic encryption. Nat. Commun. *12*, 5910. https://doi.org/10.1038/s41467-021-25972-y.

92. Chen, Y., Qin, X., Wang, J., Yu, C., and Gao, W. (2020). Fedhealth: A federated transfer learning framework for wearable healthcare. IEEE Intell. Syst. *35*, 83–93. https://doi.org/10.1109/MIS.2020.2988604.

93. Dwork, C., and Feldman, V. (2018). Privacy-preserving prediction. In Conference On Learning Theory (PMLR), pp. 1693–1702.

94. Abadi, M., Chu, A., Goodfellow, I., McMahan, H.B., Mironov, I., Talwar, K., and Zhang, L. (2016). Deep learning with differential privacy. In Proceedings of the 2016 ACM SIGSAC conference on computer and communications security, pp. 308–318. https://doi.org/10.1145/2976749.2978318.

95. Zhao, J., Chen, Y., and Zhang, W. (2019). Differential privacy preservation in deep learning: Challenges, opportunities and solutions. IEEE Access *7*, 48901–48911. https://doi.org/10.1109/ACCESS.2019.2909559.

96. Adnan, M., Kalra, S., Cresswell, J.C., Taylor, G.W., and Tizhoosh, H.R. (2022a). Federated learning and differential privacy for medical image analysis. Sci. Rep. *12*, 1953. https://doi.org/10.1038/s41598-022-05539-7.

97. Sadilek, A., Liu, L., Nguyen, D., Kamruzzaman, M., Serghiou, S., Rader, B., Ingerman, A., Mellem, S., Kairouz, P., Nsoesie, E.O., et al. (2021). Privacy-first health research with federated learning. NPJ Digit. Med. *4*, 132–138. https://doi.org/10.1038/s41746-021-00489-2.

98. Liu, H., Peng, C., Tian, Y., Long, S., Tian, F., and Wu, Z. (2022b). Gdp vs. ldp: A survey from the perspective of information-theoretic channel. Entropy *24*, 430. https://doi.org/10.3390/e24030430.

99. Pati, S., Baid, U., Zenk, M., Edwards, B., Sheller, M., Reina, G.A., Foley, P., Gruzdev, A., Martin, J., Albarqouni, S., et al. (2021b). The federated tumor segmentation (fets) challenge. Preprint at arXiv. https://doi.org/10.48550/arXiv.2105.05874.

100. Lee, J., and Kifer, D. (2021). Scaling up differentially private deep learning with fast per-example gradient clipping. In Proceedings on Privacy Enhancing Technologies. https://doi.org/10.2478/popets-2021-0008.

101. Shen, Z., and Zhong, T. (2021). Analysis of application examples of differential privacy in deep learning. Comput. Intell. Neurosci. *2021*, 4244040–4244115. https://doi.org/10.1155/2021/4244040.

102. Ficek, J., Wang, W., Chen, H., Dagne, G., and Daley, E. (2021). Differential privacy in health research: A scoping review. J. Am. Med. Inf. Assoc. *28*, 2269–2276. https://doi.org/10.1093/jamia/ocab135.

103. Jarin, I., and Eshete, B. (2022). Dp-util: comprehensive utility analysis of differential privacy in machine learning. In Proceedings of the Twelfth ACM Conference on Data and Application Security and Privacy, pp. 41–52. https://doi.org/10.1145/3508398.3511513.

104. Demelius, L., Kern, R., and Trügler, A. (2023). Recent advances of differential privacy in centralized deep learning: A systematic survey. Preprint at arXiv. https://doi.org/10.48550/arXiv.2309.16398.

105. Adnan, M., Kalra, S., Cresswell, J.C., Taylor, G.W., and Tizhoosh, H.R. (2022b). Federated learning and differential privacy for medical image analysis. Sci. Rep. *12*, 1953.

106. Malekzadeh, M., Hasircioglu, B., Mital, N., Katarya, K., Ozfatura, M.E., and Gunduz, D. (2021). Dopamine: Differentially private federated learning on medical data. Preprint at arXiv. https://doi.org/10.48550/arXiv.2101.11693.

107. Ziller, A., Usynin, D., Remerscheid, N., Knolle, M., Makowski, M., Braren, R., Rueckert, D., and Kaissis, G. (2021). Differentially private federated deep learning for multi-site medical image segmentation. Preprint at arXiv. https://doi.org/10.48550/arXiv.2107.02586.

108. Pfohl, S.R., Dai, A.M., and Heller, K. (2019). Federated and differentially private learning for electronic health records. Preprint at arXiv. https://doi.org/10.48550/arXiv.1911.05861.

109. Arasteh, S.T., Ziller, A., Kuhl, C., Makowski, M., Nebelung, S., Braren, R., Rueckert, D., Truhn, D., and Kaissis, G. (2023). Private, fair and accurate: Training large-scale, privacy-preserving ai models in medical imaging. Preprint at arXiv. https://doi.org/10.48550/arXiv.2302.01622.

110. Nasr, M., Shokri, R., and Houmansadr, A. (2018). Machine learning with membership privacy using adversarial regularization. In Proceedings of the 2018 ACM SIGSAC conference on computer and communications security, pp. 634–646. https://doi.org/10.1145/3243734.3243855.

111. Makhdoumi, A., Salamatian, S., Fawaz, N., and Médard, M. (2014). From the information bottleneck to the privacy funnel. In IEEE Information Theory Workshop (ITW 2014). IEEE, pp. 501–505. https://doi.org/10.1109/ITW.2014.6970882.

112. Jayaraman, B., and Evans, D. (2019). Evaluating differentially private machine learning in practice. In 28th USENIX Security Symposium (USENIX Security 19) (USENIX Association), pp. 1895–1912. https://doi.org/10.5555/3361338.3361469.

113. Liu, J., Oya, S., and Kerschbaum, F. (2021). Generalization techniques empirically outperform differential privacy against membership inference. Preprint at arXiv. https://doi.org/10.48550/arXiv.2110.05524.

114. Pham, Q.-V., Zeng, M., Ruby, R., Huynh-The, T., and Hwang, W.-J. (2021). Uav communications for sustainable federated learning. IEEE Trans. Veh. Technol. *70*, 3944–3948. https://doi.org/10.1109/TVT.2021.3065084.

115. Ekberg, J.-E., Kostiainen, K., and Asokan, N. (2014). The untapped potential of trusted execution environments on mobile devices. IEEE Secur. Priv. *12*, 29–37. https://doi.org/10.1109/MSP.2014.38.

116. Armato, S.G., 3rd, McLennan, G., McNitt-Gray, M.F., Meyer, C.R., Yankelevitz, D., Aberle, D.R., Henschke, C.I., Hoffman, E.A., Kazerooni, E.A., MacMahon, H., et al. (2004). Lung image database consortium: developing a resource for the medical imaging research community. Radiology *232*, 739–748. https://doi.org/10.1148/radiol.2323032035.

117. Thompson, P.M., Stein, J.L., Medland, S.E., Hibar, D.P., Vasquez, A.A., Renteria, M.E., Toro, R., Jahanshad, N., Schumann, G., Franke, B., et al. (2014). The enigma consortium: large-scale collaborative analyses of neuroimaging and genetic data. Brain Imaging Behav. *8*, 153–182. https://doi.org/10.1007/s11682-013-9269-5.

118. Karargyris, A., Umeton, R., Sheller, M.J., Aristizabal, A., George, J., Bala, S., Beutel, D.J., Bittorf, V., Chaudhari, A., Chowdhury, A., et al. (2021). Medperf: Open benchmarking platform for medical artificial intelligence using federated evaluation. Preprint at arXiv. https://doi.org/10.48550/arXiv.2110.01406.

119. Tonni, S.M., Vatsalan, D., Farokhi, F., Kaafar, D., Lu, Z., and Tangari, G. (2020). Data and model dependencies of membership inference attack. Preprint at arXiv. https://doi.org/10.48550/arXiv.2002.06856.