IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 7 September 2023, accepted 6 October 2023, date of publication 11 October 2023, date of current version 20 October 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3323932

SURVEY

Privacy and Security in Distributed Learning: A Review of Challenges, Solutions, and Open Research Issues

MUHAMMAD USMAN AFZAL[1], (Member, IEEE), ALAA AWAD ABDELLATIF[2], (Member, IEEE), MUHAMMAD ZUBAIR[3], (Senior Member, IEEE), MUHAMMAD QASIM MEHMOOD[1], (Senior Member, IEEE), AND YEHIA MASSOUD[3], (Fellow, IEEE)

[1]MicroNano Laboratory, Department of Electrical Engineering, Information Technology University (ITU), Lahore, Punjab 54600, Pakistan
[2]Department of Computer Science and Engineering, Qatar University, Doha, Qatar
[3]Innovative Technologies Laboratories (ITL), King Abdullah University of Science and Technology (KAUST), Thuwal 23955, Saudi Arabia

Corresponding authors: Yehia Massoud (yehia.massoud@kaust.edu.sa), Muhammad Qasim Mehmood (qasim.mehmood@itu.edu.pk), and Alaa Awad Abdellatif (alaa.abdellatif@ieee.org)

This work was supported by Innovative Technologies Laboratories (ITL), King Abdullah University of Science and Technology (KAUST).

ABSTRACT In recent years, the way that machine learning is used has undergone a paradigm shift driven by distributed and collaborative learning. Several approaches have emerged to enable pervasive computing and distributed learning in ubiquitous Internet of Things (IoT) systems. Numerous decentralized strategies have been proposed to deal with the limitations of centralized learning, including privacy and latency due to sharing local data, while utilizing distributed computations as a promising substitute to centralized learning. However, such distributed learning schemes come with new security and privacy concerns that should be addressed. Thus, in this paper, we first provide an overview for the emerging paradigms developed for distributed learning. Then, we performed a comprehensive survey for the privacy and security challenges associated with distributed learning along with the presented solutions to overcome them. Furthermore, we highlight key challenges and open future research directions toward implementing more robust distributed systems.

INDEX TERMS Data privacy and security, Internet of Things (IoT), deep learning, adversarial attacks.

I. INTRODUCTION

Distributed learning evolution is driven by the recent advances in the edge computing, ubiquitous Internet of Things (IoT) systems, and hardware computing capabilities. The rapid utilization of smart devices, such as self-driving cars, swarm robotics, mobile phones, wearable medical devices, and industrial IoT devices, generate an extraordinary amount of data that need to be analysed, processed, and stored. According to the statistics in [1], there will be over 75.4 billion internet-connected devices by 2025 as a result of the development of capable and affordable devices. The proliferation of interconnected devices is expected to yield a staggering volume of data, projected to reach an astonishing 181 zettabytes annually (as depicted in Figure 1). When harnessed effectively, this vast reservoir of data holds the potential to generate a substantial economic capital, estimated to reach a remarkable 11 trillion by the year 2025 [2]. Capitalizing on this extensive reservoir of data, industries are increasingly embracing Artificial Intelligence (AI)-based systems across various sectors, including robotics [3], [4], computer vision [5], and speech recognition. Renowned industry leaders are now deploying their advanced IoT services, orchestrating a transformation across diverse facets of modern existence and driving the continuous advancement of AI technologies. However, this ambitious integration presents substantial challenges in terms of memory demands and computational workloads, thereby mandating the deployment

The associate editor coordinating the review of this manuscript and approving it for publication was Zhipeng Cai.

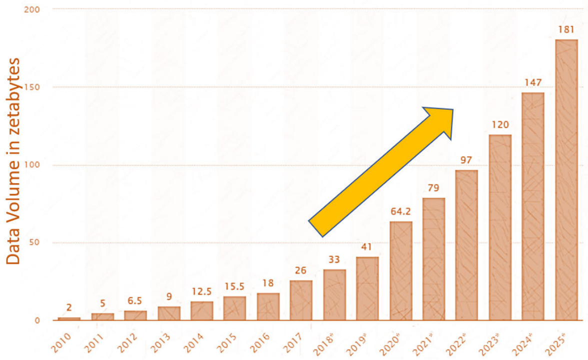


FIGURE 1. The expected data generation between 2010 and 2025 [8].

of robust servers to manage these requirements effectively. Furthermore, the accumulation and transmission of vast volumes of data to centralized servers for the training of AI or deep learning models present significant challenges that could impede the full potential of emerging intelligent systems. These challenges encompass critical aspects such as privacy preservation, latency reduction, and the management of resource-intensive computational and network burdens. Consequently, this scenario has given rise to an appealing prospect and a compelling necessity for the adoption of distributed learning methodologies.

Recently, cloud computing is no longer appropriate for real-time AI tasks due to the strict latency requirements imposed by the real-time applications and services, such as virtual and augmented reality (VR/AR) [6], and self-driving cars. For example, when it comes to autonomous cars detecting potential dangers and applying brakes, or sending data to cloud servers, meeting the latency requirements of the autonomous vehicle may not be possible. This is because, for instance, sending camera frames to remote servers requires rapid decisions, regardless of the distance and latency involved. As a result, sending data to cloud servers may not fulfill these real-time demands. The experiments in [7] revealed that it takes more than 200 ms to execute a computer vision related task having a camera frame offloaded to an Amazon server. Furthermore, in cloud-based AI schemes, privacy is a crucial concern in addition to the latency of the above-mentioned delay-sensitive applications. Due to the high risk of severe and vulnerable cyber threats such as malicious attacks and data breaches, many end-users are reluctant to transfer their private data to cloud servers. Other than privacy concerns offloading an immense amount of data to remote servers also encounters scalability and network load problems, since it might cause a bottleneck in cloud access.

Accordingly, pushing real-time AI tasks to the network edge has been proposed as a practical solution to address the challenges of latency, privacy, and scalability mentioned earlier [9], [10]. Edge devices have been improving at a fast pace to match real-time AI tasks, thanks to their physical proximity to data sources [11]. Edge devices can conduct a significant amount of computational activities without exchanging the associated data with the remote servers, ensuring agile IoT services.

Despite the promising potential of edge computing, there is still a practical limitation when attempting to run an entire AI model on a single edge device due to the constrained resources available on these devices, particularly when tasks require significant computational power. Distributed learning has recently gained significant interest as a promising solution to address the limitations of centralized learning, cloud computing, and edge computing, while also protecting data privacy and reducing the significant overhead associated with data transfer [12], [13]. In this context, each entity or user can utilize their local data to construct a local model or execute a portion of a global model, and then forward the outcomes to an orchestrator for aggregation, resulting in the final outcomes. A technique like this enables the distribution of data and AI models to be trained in a decentralized manner, while leveraging the collective power of edge devices.

A. OUR SCOPE

This survey focuses on the emerging distributed learning paradigms, which is a promising research area that closely associated with the resource limitation of participants (e.g., memory, computation, bandwidth, and energy) and the communication overheads between them. However, the process of distributed training or inference can involve a large number of participants who may communicate over wireless links, which presents new challenges related to channel capacities and conditions, delayed performance, as well as privacy and security concerns. Thus, this paper reviews the aforementioned challenges while discussing various deployed distributed learning paradigms and algorithms. The paper first identifies the motivations behind establishing distributed learning for AI applications and the related communication/network challenges. Then, it provides an overview of various privacy and security attacks that distributed learning may experience. After that, the paper discusses various defense mechanisms and algorithms presented in the literature to overcome these attacks. Finally, it presents interesting future research directions that worth further investigation. The paper's roadmap is depicted in Figure 2.

B. THE PAPER'S CONTRIBUTIONS AND STRUCTURE

The primary contributions of this paper can be summarized as follows:

1) We staged a brief background of distributed learning and introduce its different paradigms with potential applications.
2) We identify and discuss different types of privacy and security attacks related to various distributed learning paradigms.
3) We review different defensive mechanisms proposed in the literature for addressing various security and privacy attacks in different distributed learning paradigms.

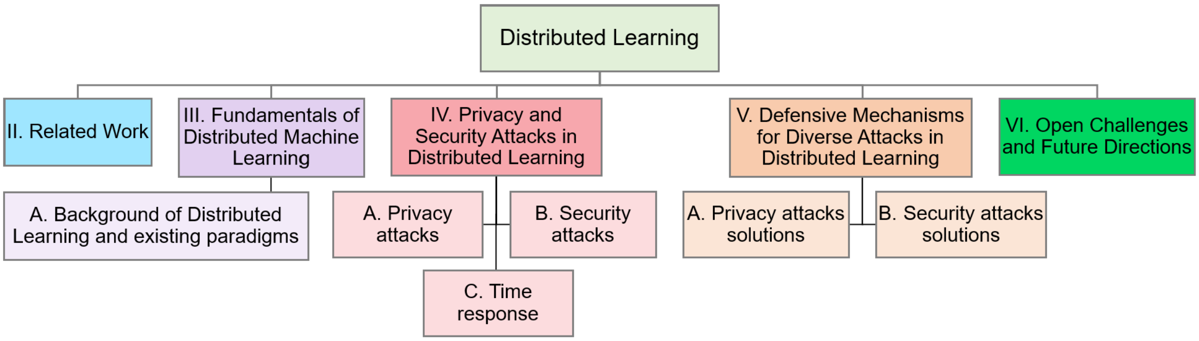

Distributed Learning
II. Related Work
III. Fundamentals of Distributed Machine Learning
A. Background of Distributed Learning and existing paradigms
IV. Privacy and Security Attacks in Distributed Learning
A. Privacy attacks
B. Security attacks
C. Time response
V. Defensive Mechanisms for Diverse Attacks in Distributed Learning
A. Privacy attacks solutions
B. Security attacks solutions
VI. Open Challenges and Future Directions

FIGURE 2. A taxonomy of the main topics presented in this survey.

4) We provide a detailed discussion of potential future challenges in this field, along with suggestions for promising research directions that merit further investigation.

The rest of this paper is arranged as follows: Section II presents the related work while highlighting the novelty of our paper. Section III introduces the background and fundamentals of distributed learning. Section IV presents diverse types of privacy and security attacks for different distributed learning paradigms, while discussing a range of defensive mechanisms proposed in the literature for addressing these attacks in Section V. Section VI provides a comprehensive discussion of the future challenges and open research directions. Finally, Section VII concludes the paper.

II. RELATED WORK

The investigation of security and privacy concerns in various distributed learning models is still in its early stages, prompting researchers to carefully scrutinize existing research and offer novel perspectives. Different studies have been presented in the field of distributed learning, including: development of algorithms and systems for distributed training of deep learning models, as well as methods and prototypes for improving the efficiency and performance of distributed learning systems [14]. The presented studies in this area focus on different issues related to partitioning and distribution of the data among different participants, implemented communication protocols between the participants during training, and strategies to combine the results from different participants to generate a final trained model or inference result [15]. In this context, the authors in [16] presented a thorough overview of privacy-preserving methods for Machine Learning as a Service, (MLaaS), beginning with traditional techniques and extending to popular deep learning techniques. In [17], the authors' emphasis was on distributed learning schemes for 5G and beyond, which spotlighted the challenges and possibilities of distributed learning in the 5G era. In [18], the authors provided a review of distributed learning techniques for 6G networks with a focus on the integration of future wireless systems with AI. On the other hand, the authors in [19] presented a systematic literature review on distributed machine learning schemes utilizing edge computing. The main objective of this work was to examine the difficulties of implementing ML/DL on edge devices in a distributed manner, with particular emphasis on the adaptation or development of techniques to operate on these resource-constrained devices. However, these two surveys lack the elaboration on privacy and security concerns. The authors in [20] provided an overview of collaborative deep learning, where they classified collaborative learning schemes into direct, indirect, and peer-to-peer approaches, while highlighting some of their associated privacy concerns. Furthermore, the authors examined general cryptographic algorithms and other techniques that can be utilized for privacy preservation and highlighted their advantages and disadvantages in the collaborative learning setting. In [21], the authors investigated the possible threats of deep learning concerning black and white box attacks, and discussed the relevant countermeasures for both offensive and defensive purposes. In [22], a comprehensive survey on the integration of differential privacy with ML was presented, commonly referred to as differentially private ML. The authors categorized the related works into two primary groups, based on diverse differential privacy mechanisms: the Laplace/Gaussian/exponential mechanisms and the output/objective perturbation mechanisms.

There are some related surveys that tackled security and privacy concerns in Federated Learning (FL) and discussed corresponding solutions [2], [23], [24]. For instance, the authors in [25] provided a review of current literature on FL, where they presented a functional architecture for FL systems along with the related techniques. Moreover, they discussed FL systems from four perspectives: different types of parallelism, aggregation algorithms, data communication, and security. The authors in [24] presented a review and classification of threat models in of FL, along with two major types of attacks that can occur in FL, i.e., poisoning attacks and inference attacks. This review provided a valuable resource for individuals looking to gain an understanding of FL and its potential privacy concerns, while emphasizing on the underlying assumptions and essential techniques used in different attacks. In [26], the primary focus was on the application and security of FL in healthcare applications. The authors discussed various architectures and models of FL in the context of healthcare, and provided a comprehensive overview of the use cases of FL in this domain. The authors highlighted the potential benefits of using FL in healthcare, such as improving data privacy and security, enabling collaboration between multiple healthcare organizations, and enhancing the accuracy of medical diagnoses and treatments.

While there have been previous surveys in the literature on the topic of distributed machine learning, each providing valuable insights and information (as summarized in Table 1), our survey distinguishes itself by focusing on recent studies that have integrated both security and privacy concepts in the context of distributed machine learning. Thus, this survey aims to fill a gap in the literature by examining the most recent developments in security and privacy issues in distributed machine learning. It provides a comprehensive review of the current state of the field and highlight the emerging trends and challenges in this area. In particular, we conduct an in-depth review of the fundamental attacks that occur in distributed learning and identified effective solutions that have made this approach practical and effective across a variety of applications. By delving deep into these attacks and solutions, we aim to provide a clear understanding of the underlying principles that have enabled the success of distributed learning. Additionally, we explore various methodologies and solutions that have been proposed in the literature, which can serve as a valuable resource for researchers to expand their knowledge and inspire the development of new strategies for distributed learning. Thus, this survey offers a fresh perspective on the intersection of security and privacy in distributed learning, making it a valuable resource for researchers and practitioners in this field.

III. FUNDAMENTALS OF DISTRIBUTED MACHINE LEARNING

Distributed learning is a powerful approach that involves training machine learning models using data and resources that are distributed across multiple devices or users. This approach is particularly useful for training large and complex models that would be too difficult to train on a single machine. Additionally, distributed learning can enhance the efficiency and scalability of the learning process. Compared to traditional centralized machine learning approaches, distributed learning provides several benefits. These include enhanced scalability, efficiency, and accuracy of machine learning models, along with reductions in training time. Additionally, distributed learning allows for improved model performance through multiparty collaboration and the ability to feed distributed data to the model. Thus, it is becoming increasingly popular due to the vast amounts of data generated by modern applications and the need for efficient processing of this data.

Despite the benefits of distributed learning, it also presents several challenges that must be addressed. One such challenge is communication overhead, which arises due to the involvement of multiple devices in the learning process that need to communicate with one another. Additionally, security and privacy concerns must be addressed to ensure that sensitive data remains protected. Fault tolerance is another issue that must be considered, as the failure of any individual device can disrupt the learning process. Finally, guaranteeing convergence, or the ability of the model to converge on a solution, is also a challenge that must be addressed in distributed learning.

A. BACKGROUND OF DISTRIBUTED LEARNING AND EXISTING PARADIGMS

Distributed learning, a subset of AI's core technologies, involves training models on data distributed across multiple devices or participants. The concept of distributed learning becomes an increasingly popular approach in the era of big data and AI due to its ability to accelerate the training process. By distributing the data across multiple devices, each device can work on a different subset of the data simultaneously, reducing the overall training time. This approach is particularly useful for large datasets or complex models that may take a considerable amount of time to be trained. In addition to faster training times, distributed learning also offers greater scalability. As the amount of data grows or the models become more complex, distributed learning can easily handle the increased workload by adding more devices or participants to the learning process. This allows for the training of models on extremely large datasets that would be impractical to manage on a single machine. However, implementing distributed learning comes with its own set of challenges. Communication between devices and maintaining model consistency can be difficult, and failures of individual devices can cause significant disruptions. Researchers have developed various techniques to address these issues, including fault tolerance strategies such as checkpointing and replicated training [28], [29], and model consistency techniques such as parameter averaging and consensus optimization [30].

There are several paradigms of distributed learning that differ in how the data and computations are distributed across the participants [31]. These paradigms include:

- Data parallelism: In this paradigm, the data is partitioned across multiple participants in a distributed manner, and each participant performs computations on its local data. The intermediate results are then exchanged among the participants to update/generate the final model.
- Model parallelism: In this paradigm, the model is partitioned across multiple participants in a distributed system, and each participant is responsible for executing a part of the model's output. The intermediate results are then exchanged among the participants to generate the final output. This paradigm is useful for large models that cannot be trained on a single machine due to memory or computational constraints (see Figure 3).
- Hybrid parallelism: This paradigm combines data parallelism and model parallelism, in which each participant has access to a subset of the data, and the computations are distributed across the participants.
- Federated Learning (FL): In this paradigm, the data is distributed across multiple devices or participants, and

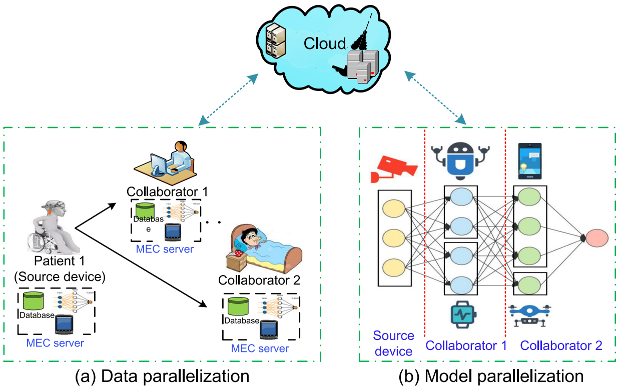
TABLE 1. Related surveys on privacy and security in distributed learning.

Ref.	Topic	Privacy	Security	Use cases	Limitations
[17]	Distributed machine learning for 5G and beyond	✓	✓	Challenges and opportunities of distributed machine learning in 5G era with an emphasis on Communication and optimization aspects.	No explicit focus on security and privacy issues.
[20]	A review on distributed deep learning and associated privacy preservation techniques.	✓	✗	Characterized collaborative deep learning during the training stage and subdividing it into direct, indirect, and peer-to-peer networks; discussed privacy-preserving strategies and related concerns, and studied related privacy issues, such as weight update theories and homomorphic encryption for privacy preservation in collaborative learning	There was minimal emphasis on the security aspect.
[25]	A review of current literature on federated learning	✓	✓	Discussed functional overview of FL systems, distributed training, and data manipulation; presented the life cycle of FL models, different aggregation algorithms.	FL was the main focus of this work.
[27]	Review of privacy-preserving distributed learning from federated databases in healthcare domain	✓	✗	Discussing the potential of distributed learning in facilitating big data for medical applications. The main objective was to conduct a review of the major implementations of distributed learning in healthcare.	No explicit focus on security issues.
[18]	Overview of distributed learning techniques for 6G networks	✗	✗	Investigating the integration of AI with wireless systems to enhance the performance and capabilities of wireless 6G networks.	There is a lack of elaboration on security and privacy issues in distributed settings.
[19]	A literature review on distributed learning leveraging edge computing	✗	✗	Discussing the challenges of distributed learning on edge devices, focusing on techniques for caching, training, inference, and offloading adapted for these devices' restrictions.	Security and privacy issues in distributed learning settings have not been sufficiently elaborated on.
[26]	A security review is being conducted on the application of federated learning in healthcare	✗	✓	Discussing various algorithmic architectures and classification models used in federated learning, with a specific emphasis on their applications in healthcare.	No explicit focus on security and privacy issues of different distributed learning paradigms.

the training is performed locally on each device. Then, the local model updates are forwarded to a central server, which aggregates them to update the global model [32]. This paradigm aims at enhancing the privacy because it allows the data to remain on the devices and only sends the model updates through the network [33]. FL comes with two modes: Synchronous FL and Asynchronous FL. The former synchronizes updates, ensuring global model consistency but introducing potential latency. The latter mode allows independent updates, accommodating variable participation and network conditions but risking model inconsistency.

- Ensemble learning: In this paradigm, multiple models are trained independently on different subsets of the data, and their predictions are combined to produce the final output.

We remark that these paradigms of distributed learning have different strengths and weaknesses, and the choice of paradigm depends on the available resources and applications' requirements.



(a) Data parallelization (b) Model parallelization

FIGURE 3. Data/model parallelization in distributed learning.

There are various configurations of distributed learning, which vary based on how the data and computations are distributed across the participants. In some configurations, a central server, i.e., Remote agents (RA), may be responsible for aggregating the model updates, while in others, the updates may be exchanged directly between the participants. RA and work agents (WA) are key components of many distributed learning configurations. RA is responsible for coordinating the WA's work and ensuring that the model updates are aggregated correctly. The WA performs the actual computations on the local data and sends the updates to the RA for aggregation. Hence, distributed learning configurations can be categorized into three main groups (see Figure 4): hierarchical distributed learning (or master-slave configuration), centralized distributed model, and fully distributed learning (in which every participant interacts independently with other participants). We remark that different averaging and topology schemes can be used to combine the model updates from different WAs, such as weighted averaging, local updates, and gossip-based protocols [34]. Moreover, the choice of distributed learning configuration depends on the specific application and requirements, including the size of the dataset, the computing resources available, and the privacy concerns.

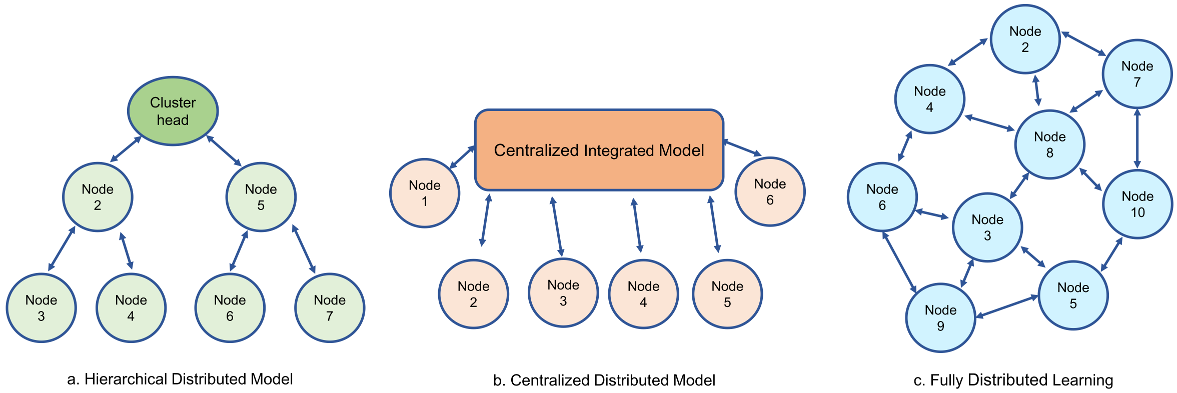


FIGURE 4. Examples for some of the existing distributed learning configurations.

B. LESSONS LEARNED

Based on the conducted review of the development and implementation of distributed learning, several key lessons can be concluded:

1) Scalability: As the size of datasets and complexity of models continue to grow, it is crucial to be able to scale learning algorithms across multiple participants. Distributed learning offers a promising approach to achieve this scalability through leveraging data parallelism and model parallelism, which can help prevent bottlenecks in the training process.
2) Fault tolerance: In distributed systems, failures of individual devices or users can cause significant disruptions to the learning process. Therefore, it is essential to design systems with fault tolerance, using techniques such as checkpointing and replicated training.
3) Communication overhead can be a bottleneck: In distributed learning, communication between diverse participants can become a bottleneck, slowing down the training process. Various strategies have been proposed to address this issue, such as using compression techniques to reduce the size of data transferred between participants [35].
4) Model consistency: A key aspect of distributed learning is achieving convergence and consistent model updates across multiple devices. It is important to ensure that all participants have access to consistent and up-to-date data/models. Techniques such as distributed stochastic gradient descent, parameter averaging, consensus optimization can help ensure that models remain consistent across diverse participants [30].
5) Hardware and software heterogeneity: In distributed learning, participants may have different hardware and software configurations, which can make it challenging to achieve optimal performance. Various techniques have been proposed to address this issue, such as adjusting learning rates and partitioning data based on device characteristics.

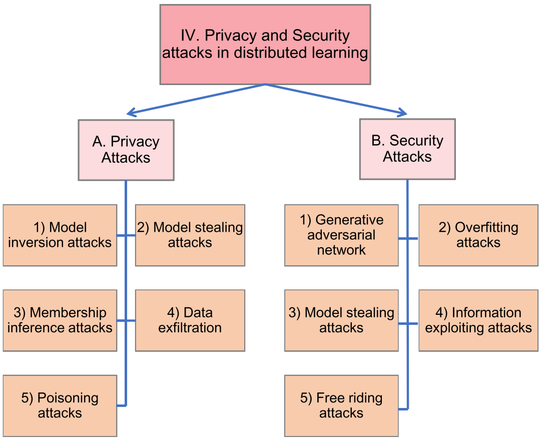

IV. Privacy and Security attacks in distributed learning
A. Privacy Attacks
B. Security Attacks
1) Model inversion attacks
2) Model stealing attacks
3) Membership inference attacks
4) Data exfiltration
5) Poisoning attacks
1) Generative adversarial network
2) Overfitting attacks
3) Model stealing attacks
4) Information exploiting attacks
5) Free riding attacks

FIGURE 5. Privacy and security attacks discussed in this survey.

IV. PRIVACY AND SECURITY ATTACKS IN DISTRIBUTED LEARNING

This section introduces and categorizes the key privacy and security threats that have arisen in the context of distributed learning, while discussing different factors that affect their time response (see Figure 5).

A. PRIVACY ATTACKS

Privacy attacks in distributed learning refer to actions or tactics that can be employed to gain unauthorized access to sensitive information from a machine learning model or the data used to train the model. These attacks can be conducted by malicious actors seeking to obtain confidential or sensitive data for nefarious purposes or by legitimate users who may inadvertently compromise the privacy of the data. In what follows, we will explore some of the common types of privacy attacks that have been identified in the context of distributed learning.

1) MODEL INVERSION ATTACK

Model inversion attacks, also known as inverse machine learning attacks, involve using a machine learning model to generate synthetic data that closely resembles the original data used to train the model. These attacks can be used to infer sensitive information about individuals represented in the data, such as their personal characteristics, habits, or preferences [36]. The idea behind model inversion attacks is to use the machine learning model as a tool to generate synthetic data that accurately represents the underlying patterns in the original data. By analyzing the synthetic data, an attacker can potentially deduce information about the individuals represented in the original data, even if that information was not explicitly included in the original data. Model inversion attacks can be particularly concerning in the context of sensitive data, such as medical records, financial data, or other types of personal information.

2) MODEL STEALING ATTACK

From a privacy perspective, model stealing attacks can have implications related to the unauthorized extraction of sensitive information.The attack could be considered a privacy breach. When an adversary successfully steals a machine learning model [37], they might gain insights into the underlying training data, potentially exposing sensitive patterns or characteristics of the data. This breach of privacy could lead to the exposure of proprietary information, trade secrets, or personally identifiable information that was used in the model's training.

3) MEMBERSHIP INFERENCE ATTACKS

They refer to a class of attacks that aim to determine whether a specific individual's data was used to train a ML model [38]. In other words, the goal of a membership inference attack is to determine if a particular individual is a member of the dataset that was used to train the model. Membership inference attacks can be carried out by analyzing the output of the ML model, and attempting to infer whether a specific data point was used in the model's training. This can be done by comparing the model's predictions on the target data point to its predictions on a separate set of data points that were not used in training. If the model's predictions on the target data point are significantly more accurate than its predictions on the other data points, it can be inferred that the target data point was likely used to train the model.

4) DATA EXFILTRATION

This type of attack involves accessing and stealing sensitive data from one of the participating parties in a distributed learning system [39]. This can be done through direct access to the data or by exploiting vulnerabilities in the shared machine learning model.

5) POISONING ATTACKS

This type of attack aims to introduce malicious or misleading data into the training process of a ML model. The goal of such an attack is to cause the model to make incorrect predictions or inferences when deployed in the real world [40]. Specifically, in a poisoning attack, a malicious user may inject a small number of data points with incorrect labels into the training data, with the aim of causing the model to learn a biased or incorrect decision boundary [41]. Alternatively, a larger number of data points may be modified in a subtle way that is designed to influence the model's behavior in a particular direction. These modifications can be carried out in a targeted or indiscriminate manner, depending on the attacker's goals. Poisoning attacks can be particularly challenging to detect and mitigate, as the malicious data points may be indistinguishable from legitimate data points. Additionally, since the model is trained on a large amount of data, it may be difficult to identify and remove the poisoned data points without negatively impacting the overall performance of the model.

B. SECURITY ATTACKS

In distributed learning, security attacks are actions or strategies that can be used to compromise the security of a ML model or the data used to train it. These attacks can take many forms, and can range from simple data breaches to more sophisticated attacks such as model stealing attacks. Some of the most common types of security attacks in the context of distributed learning include:

1) GENERATIVE ADVERSARIAL NETWORK (GAN) ATTACKS

A GAN is a type of deep learning model that comprises two neural networks: a generative network and a discriminative network [42]. The generative network generates synthetic data that is similar to the real data, while the discriminative network learns to distinguish between real and synthetic data [43]. A GAN-based attack is a type of adversarial attack that can be employed against a ML model in the context of distributed learning [44]. In this type of attack, the attacker trains a generative model, such as a GAN (Generative Adversarial Network), to generate synthetic data that is similar to the data used to train the target model [45]. The attacker can then use this synthetic data to deceive or attack the target model in various ways, causing it to make incorrect predictions or inferences [46]. For example, an attacker can use the synthetic data generated by the GAN to attack the target model in a targeted manner. They can create synthetic data that is designed to force the target model to make specific incorrect predictions or inferences. Additionally, an attacker can use GANs to launch a poisoning attack by generating synthetic data that contains malicious inputs. When this data is used to train the target model, it can lead to the model being poisoned, causing it to make incorrect predictions or inferences. Therefore, GAN-based attacks have been shown to be effective at causing ML models to make incorrect predictions or inferences, and they can be a powerful tool for adversarial attackers.

2) OVERFITTING ATTACKS

This attack involves training a machine learning model on a small, carefully selected dataset to cause the model to perform poorly on other data [47]. It can be used to undermine the performance and accuracy of the model, leading to incorrect predictions and compromised results.

3) MODEL STEALING ATTACKS

From a security standpoint, model stealing attacks can be seen as a form of intellectual property theft and a potential vulnerability in machine learning systems [37]. If an attacker manages to replicate a model, they could use it for malicious purposes, such as crafting adversarial attacks to exploit vulnerabilities in the model or using it for unauthorized purposes without proper authorization. This undermines the security of the machine learning system and could lead to various security risks, especially if the replicated model is used for harmful activities.

4) INFORMATION EXPLOITING ATTACKS

These attacks involve attempts to extract sensitive information about individuals or institutions by analyzing the predictions made by a ML model [48]. They can pose a significant threat to the privacy and security of the individuals or organizations whose data was used to train the model. By exploiting the patterns and predictions generated by the model, attackers can infer sensitive information about the training data, such as the demographics, behaviors, or preferences of the individuals or organizations. Information exploiting attacks can also compromise the security of the model itself, leading to incorrect or biased predictions that can be manipulated by attackers.

5) FREE RIDING ATTACK

A free-riding attack occurs when someone exploits a ML system's resources and capabilities without providing any data or resources themselves, leading to potential overuse and degradation of the system's resources. This kind of attack can be a problem in distributed ML systems, as it can reduce performance and accuracy for all users and increase maintenance costs. It's even more critical if the model is confidential or has sensitive information, as the attacker may gain unauthorized access to exploit it.

Several types of free-riding attacks can be carried out against distributed ML systems. Some common examples include:

1) Data free riding: This form of intrusion entails gaining access to and utilizing the data of a ML system without providing any of one's own data.
2) Model free riding: This attack refers to the utilization of a trained model without contributing to the model's training or maintenance. It can be carried out by downloading or accessing the model and utilizing it for personal objectives.
3) Infrastructure free riding: This type of attack involves using the infrastructure of a ML system (such as computing resources or storage) without contributing to the maintenance or development of the system.

C. TIME RESPONSE

Time response of privacy and security attacks in distributed learning refers to how quickly an adversary can gather sensitive information or perform malicious activities based on the data or models used in the distributed learning process. Hence, it assesses the speed at which privacy/security breaches or attacks can take place. It is important to note that a rapid response time in privacy/security attacks could lead to more significant risks, as adversaries could exploit vulnerabilities before proper countermeasures can be implemented [49]. Thus, when designing secure distributed learning systems, it is crucial to consider not only the effectiveness of the proposed defense mechanisms but also the potential speed at which attacks can occur.

Several factors can affect the response time of various attacks, depending on: learning model complexity, attack objective, available resources at the attacker's disposal, attack strategy, data quality and quantity, defensive measures, and knowledge of model architecture. For instance, more complex models might require additional time [50] for inversion attacks due to the increased difficulty of modeling their internal behavior. Moreover, the specific goal of the attack can influence the response time. Different attacks might target various aspects of the model, such as generating adversarial examples or undermining the model's training process. Furthermore, the computational resources available to the attacker can influence the speed of the attack [51]. More powerful hardware or distributed computing can expedite the inversion process. In terms of attack strategy, the chosen approach by the attacker can impact the time response. Some attacks may prove to be more efficient than others. Moreover, the quality and quantity of available data for the attack can affect its speed. Having more data might result in more accurate inferences. Finally, defensive measures and knowledge of the model architecture can also significantly impact the response time. If the target model is safeguarded with effective privacy-preserving techniques or security mechanisms, the response time of an attack might increase, as it becomes more challenging for the attacker to deduce accurate information. On the contrary, if the attacker possesses knowledge of the architecture and details of the model being targeted, it can expedite the attack.

V. DEFENSIVE MECHANISMS FOR DIVERSE ATTACKS IN DISTRIBUTED LEARNING

In this section, we present various defensive mechanisms for diverse privacy and security attacks that have been discussed above.

A. PRIVACY ATTACKS SOLUTIONS

In what follows, we discuss various defensive mechanisms for privacy attacks that can be implemented in distributed learning (see Figure 6).

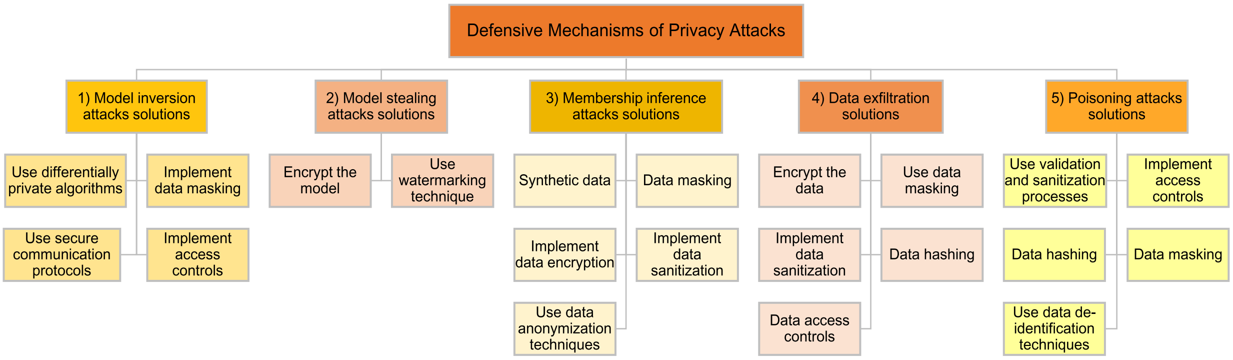

Defensive Mechanisms of Privacy Attacks
1) Model inversion attacks solutions
Use differentially private algorithms
Implement data masking
Use secure communication protocols
Implement access controls
2) Model stealing attacks solutions
Encrypt the model
Use watermarking technique
3) Membership inference attacks solutions
Synthetic data
Data masking
Implement data encryption
Implement data sanitization
Use data anonymization techniques
4) Data exfiltration solutions
Encrypt the data
Use data masking
Implement data sanitization
Data hashing
Data access controls
5) Poisoning attacks solutions
Use validation and sanitization processes
Implement access controls
Data hashing
Data masking
Use data de-identification techniques

FIGURE 6. The discussed defensive mechanisms for privacy attacks in distributed learning.

1) MODEL INVERSION ATTACKS

Several methods have been presented in the literature to mitigate model inversion attacks in distributed ML systems, including:

a: DIFFERENTIALLY PRIVATE ALGORITHMS

Differential privacy is a technique used to protect the privacy of individuals while sharing or releasing data [52]. By adding noise or randomness to the data, differential privacy makes it harder for attackers to identify specific individuals or sensitive information in the data. The utilization of differentially private algorithms, as exemplified in [53] and [54], can increase the difficulty for potential attackers to reverse-engineer confidential data from the model. Indeed, differential privacy can obtain statistical guarantees about the privacy of participants, however it does not protect against all possible attacks. For example, an attacker may be able to infer sensitive information about individuals by combining the differentially private output with other publicly available data or by using sophisticated ML models to analyze the output. Moreover, the level of privacy protection provided by differential privacy may depend on the specific implementation and configuration of the algorithm.

b: DATA MASKING

Data masking mechanisms refer to substituting sensitive data with a random or synthetic version of the original data that retains the statistical characteristics of the original data but does not disclose any sensitive information. The purpose of data masking mechanisms, as described in [55], [56], and [57], is to minimize the risk of exposing sensitive information while still allowing access to the data for analysis or other purposes. Indeed, by using data masking techniques, attackers will have a more challenging task for reverse-engineering sensitive data from the shared models. Since the data masking techniques alter the original data, an attacker cannot rely on a direct mapping of the original data to extract sensitive information. Instead, they would have to employ more sophisticated methods to infer the sensitive data, which can be significantly more difficult and resource-intensive. However, the effectiveness of data masking mechanisms depends on the specific method used and the strength of the randomization or synthesis used in the masking process. In some cases, an attacker may still be able to infer sensitive information from the masked data, particularly if they have prior knowledge or access to other data sources. Thus, it is important to evaluate and select the appropriate data masking mechanism based on the specific context and the potential threats and risks to the data.

c: SECURE COMMUNICATION PROTOCOLS

Implementing secure communication protocols, such as transport layer security (TLS) [58], [59], [60] or secure sockets layer (SSL) [61], is beneficial in safeguarding the model from inversion attacks by preventing unauthorized access to the model and the data it handles.

d: PROPER ACCESS CONTROLS

The implementation of efficient access controls mechanisms [62], including authentication, authorization, and related protocols, is an effective measure to prevent unauthorized access to the shared models and associated data.

2) MODEL STEALING ATTACKS

To prevent/mitigate the impact of model stealing attacks, it is important to employ robust security measures such as access controls, network segmentation, and secure communication protocols. Additionally, implementing techniques such as watermarks or other forms of model fingerprinting can help identify stolen models and deter potential attackers. Finally, monitoring for suspicious activity and promptly responding to security incidents can help in minimizing the impact of model stealing attacks. Other than secure communication and access controls, following ways also presented in the literature to mitigate the model stealing attacks.

a: ENCRYPTING THE MODEL

Encrypting the trained model can make it more difficult for attackers to access and copy the model without permission [63], [64].

b: USING WATERMARKING TECHNIQUE

Watermarking techniques [65], [66] involves the addition of a unique identifier to the model that can be used to trace the origin of the model if it is accessed or used without permission. Furthermore, a combination of mentioned security measures can help in mitigating the risk of model stealing attacks in distributed learning systems.

3) MEMBERSHIP INFERENCE ATTACKS

To mitigate the risk of membership inference attacks, it is important to employ effective privacy-preserving techniques, such as differential privacy or secure multi-party computation. Additionally, it may be necessary to limit access to the data used to train the model and to implement efficient access controls mechanisms to prevent unauthorized access to the models' output. In what follows, we summarize the key mechanisms presented in the literature for mitigating the risk of membership inference attacks:

a: LEVERAGING DIFFERENTIALLY PRIVATE ALGORITHMS

These algorithms rely on adding noise to the training data while protecting the privacy of the people whose data are being utilized, allowing for accurate model training. It may be more challenging for attackers to deduce the membership of particular data in the training set when differentially private algorithms are used.

b: USING SYNTHETIC DATA

Synthetic data [67], [68] refers to the data that is artificially generated to mimic the statistical properties of real data [69], but does not contain any sensitive information. Using synthetic data helps in protecting the privacy of the individuals whose data is being used in the training, while still allowing for accurate model training.

c: IMPLEMENTING DATA ENCRYPTION SCHEMES

Encrypting the training data or the trained model makes it more difficult for the attackers to access and use the data or model for membership inference.

d: ADOPTING DATA SANITIZATION

Data sanitization [70], [71] involves removing or replacing sensitive information from the training data in order to protect the privacy of the participating users in the learning [72].

e: DEPLOYING DATA ANONYMIZATION TECHNIQUES

Data anonymization [73] techniques involve removing or replacing identifying information from the training data in order to protect the privacy of the participating users [74].

It is worth noting that data masking schemes can also serve the purpose of mitigating membership inference attacks.

4) DATA EXFILTRATION ATTACK

To mitigate the risk of data exfiltration attacks, it is important to implement robust security measures such as access controls, network segmentation, and secure communication protocols, while limiting the access to the data used in the distributed learning and monitoring suspicious activity. Furthermore, employing techniques such as differential privacy or homomorphic encryption can help in protecting the privacy of the data and mitigate the impact of any potential data breaches. To sum up, we recap the main presented mechanisms to overcome this attack as follows:

a: DATA ENCRYPTION

Encrypting the data that is processed by the ML model can make it more difficult for the attackers to exfiltrate sensitive data [75], [76].

b: DATA ANONYMIZATION TECHNIQUES

These techniques involve removing or replacing identifying information from the data that is processed by the ML model to protect the privacy of the participants.

c: DATA HASHING

It involves replacing the original data with a unique code, or hash, that is derived from the data [77], [78]. This can make it more difficult for the attackers to exfiltrate sensitive data, as the original data cannot be reconstructed from the hash.

d: DATA ACCESS CONTROLS

Implementing data access controls, such as data access policies and data segregation techniques, helps in preventing unauthorized access to sensitive data.

It is crucial to emphasize that the utilization of both data masking and data sanitization schemes can be instrumental in proficiently mitigating the risks associated with data exfiltration.

5) POISONING ATTACKS

To mitigate the risk of poisoning attacks, it is important to employ robust security measures such as access controls, network segmentation, and secure communication protocols. It may be also necessary to limit the access to the data used in the training process and to implement data validation techniques to detect and remove any suspicious data points. Moreover, employing techniques such as federated learning, where the data is kept decentralized and training is done on local devices, can help in reducing the impact of poisoning attacks. Several mechanisms have been presented in the literature to deal with the threats of poisoning attacks [79] in distributed learning systems, which include:

(i) Leveraging robust validation and sanitization processes to ensure that the acquired data are free of malicious inputs. This can include implementing data quality checks, such as checking for missing values or outliers, and removing or replacing data that does not meet certain criteria.

(ii) Implementing data de-identification techniques which involve removing or obfuscating identifying information, such as names, addresses, and social security numbers, so that the participants in distributed learning cannot be directly identified from the data [80], [81]. However, it is important

to note that de-identification is not foolproof and can be circumvented through re-identification attacks. Therefore, it is crucial to use additional privacy protections, such as access controls, encryption, and monitoring, to further safeguard sensitive data.
(iii) Leveraging data anonymization techniques which involve removing or replacing identifying information from the training data in order to protect the privacy of the individuals whose data is being used.
(iv) The use of both data masking and data sanitization can also assist in minimizing the risks associated with poisoning attacks.

B. SECURITY ATTACKS

In the next discussion, we will explore diverse protective mechanisms that can be utilized in distributed learning to counter security threats (see Figure 7).

1) GAN-BASED ATTACKS

GAN-based attacks in distributed learning can be countered with several defensive mechanisms, which include: data augmentation, adversarial training, robust optimization, differential privacy, model distillation, and federated learning. Data augmentation, i.e., one way to make a GAN-based attack less effective is to enhance the diversity of the training data [82]. This can be achieved through data augmentation [83], [84] techniques, such as cropping, rotating, and adding noise to the acquired data. Another approach is to use adversarial training [85], in which the model is trained on a mix of real and generated data [86], [87]. The goal of adversarial training is to improve the model's ability to distinguish between real and generated data, thereby making it more resistant to GAN-based attacks. Defense-GANs schemes [88] are a special type of GAN that are designed specifically to defend against GAN-based attacks. They can be trained to recognize and reject generated data, making them an effective countermeasure against GAN-based attacks. Model distillation schemes are also presented to train a smaller, more efficient model using the output of a larger, more complex model [89]. This can help defend against GAN-based attacks by reducing the number of possible attack points. Ensemble methods are used also for defending against GAN-based attack [90], [91], [92], where multiple models are trained and their predictions are combined to create a final prediction. Due to the possibility that the data generated may not be consistent across various models, this is useful in the case of GAN-based attacks. Some GAN-based attacks rely on manipulating the input data before it is fed into the learning model. Hence, by leveraging input preprocessing schemes [93], the manipulated input data can be detected and removed to reduce the effectiveness of the GAN-based attacks.

2) OVERFITTING ATTACKS

To prevent overfitting attacks in distributed learning, it is important to take the following precautions:

- Using a simple model structure: A complex model structure with a large number of parameters is more prone to overfitting. Using a simpler model structure can help in reducing the risk of overfitting.
- Leveraging a large dataset: A large dataset can provide more information to the learning model, helping it to learn generalizable patterns rather than memorizing the training data.
- Regularizing the model: Regularization techniques [94], [95], such as L2 regularization or dropout can help prevent overfitting by introducing additional constraints on the model parameters.
- Deploying cross-validation: Cross-validation involves dividing the dataset into training and validation sets, and using the validation set to evaluate the model's performance. This can help in identifying the overfitting and allow for adjustments to be made to the model.
- Using early stopping criteria: Early stopping entails monitoring the model's performance on the validation set and stopping the training when the model's performance begins to degrade. This can help in preventing the overfitting by stopping the training before the model becomes too closely fit to the training data.
- Implementing transfer learning: Transfer learning [96], [97] involves using a pre-trained model as the starting point for the training of a new model. This can help in decreasing the risk of overfitting by leveraging the knowledge learned from the pre-trained model.
- Leveraging bagging techniques: Bagging is a technique that applies training multiple models on various subsets of the training data and combining their predictions. This can help improve the generalization of the model by reducing the variance of the predictions.
- Using boosting scheme: It entails training a series of models, each of which is trained to fix the mistakes made by the model before it, actually lowering the bias of the predictions, this can help in the model's generalization.

3) MODEL STEALING ATTACKS

Different schemes have been presented in the literature to protect against model stealing attacks in distributed learning. These schemes rely on implementing one or more of the following measures. First, deploying the learning models in a secure environment, such as a secure server or cloud-based platform, can help prevent unauthorized access or copying of the model. Second, implementing model watermarking which involves adding a unique identifier or "watermark" to the model, in order to help in identifying the source of any copied models. Third, leveraging model obfuscation [98] which involves making the model more difficult to understand or reverse-engineer. Fourth, implementing model versioning that helps the participants to keep track of versions of the model, which is helpful in implementing controls to ensure that only authorized users have access to the latest version, hence preventing the use of outdated or stolen models. Fifth, by regularly updating [99] the model with new data and retraining, it is possible to prevent the use

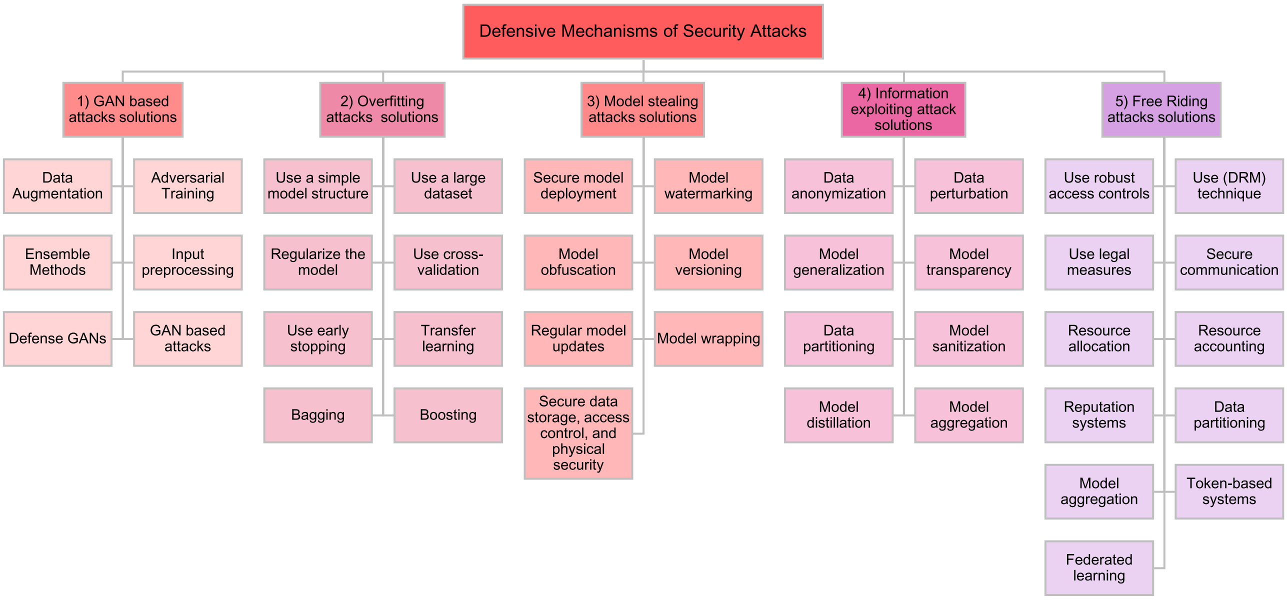


FIGURE 7. The discussed defensive mechanisms for security attacks in distributed learning.

of outdated or stolen models while improving the model's accuracy. In addition to that, model wrapping [100] is also used to encapsulate the model in a secure wrapper that controls access to the model and logs any attempts to access it. Finally, secure data storage, access control, and physical layer security [101] are also utilized to prevent unauthorized access. The shared learning models can be protected from unauthorized access by considering physical layer security measures, as well as secure servers and data centers. Indeed, leveraging access control measures like user authentication and authorization allow for preventing unauthorized access to the model, while encrypting the data used to train the model can help also in protecting it from being decrypted without authorization.

4) INFORMATION EXPLOITING ATTACKS

To protect against these types of attacks, it is important to use secure and robust techniques in the design and implementation of distributed learning systems. This calls for deploying security initiatives such as encryption, secure communication protocols, and other protective methods in order to prevent unauthorized access or misuse of learning data and models. The main presented schemes to tackle information exploiting attacks in distributed learning are:

- Data anonymization: Anonymizing the acquired data [73], [74] can help in preventing the identification of participants and protect their privacy.
- Data perturbation: Perturbing the used data [102], [103] to train the model, by adding noise or applying transformations to the data, can help in preventing the extraction of sensitive information.
- Model generalization: Designing the learning models to be more generalizable [104], rather than closely fit to the training data, allows for preventing the extraction of sensitive information from the model's predictions.
- Model transparency: Implementing techniques such as model interpretability or explainability enables the model's predictions to be more transparent while turning it to be more difficult for the attackers to extract sensitive information.
- Model sanitization: Sanitizing the model, by removing sensitive features or applying perturbations to the model parameters, can help in avoiding the information exploiting attacks.
- Model distillation: Model distillation [105], [106], [107], [108] involves training a smaller, simpler model that is able to reproduce the predictions of a larger, more complex model. This can help in preventing the extraction of sensitive information from the model's predictions.
- Model aggregation: Model aggregation [109] involves training multiple models and combining their predictions to make a final prediction, which improves the generalization of the model and makes it more difficult for the attackers to extract sensitive information.
- Data partitioning schemes can also be employed to mitigate information-exploiting attacks.

5) FREE RIDING ATTACK

To mitigate the risk of free-riding attacks, it is important to implement appropriate access controls and resource allocation policies. This can include setting limits on the portion of resources that can be accessed by each user or implementing payment or credit systems to ensure that the participating users contribute to the system in proportion to their use of its resources. There are several ways in which we can mitigate the risk of free-riding attacks in distributed learning, which include:

- Using robust access controls: Implementing robust access controls, such as authentication and authorization protocols, can help in preventing unauthorized access to the model and the data it processes. This can include measures such as user authentication, role-based access controls, and data encryption.
- Leveraging digital rights management (DRM) techniques: DRM techniques [110], [111] involve using technical measures to control access/use of the ML models. This can include measures such as encryption, digital keys, and license management systems.
- Utilizing legal measures: The participating users in distributed learning systems can use legal measures, such as copyright law and contracts, to protect their ML models and prevent unauthorized use or replication.
- Deploying secure communication protocols: Implementing secure communication protocols can help in avoiding the malicious participants from intercepting or manipulating data or model parameters during the training process.
- Implementing efficient resource allocation schemes: Implementing mechanisms for fairly allocating resources, such as CPU or GPU time, among different participants can help in preventing free riding attacks by ensuring that all participants contribute resources to the training process.
- Resource accounting: Keeping track of the resources contributed by each participant, such as data or computing resources, can help in identifying and avoiding free riding attacks through ensuring that all participants contribute fairly to the training process.
- Reputation schemes: Implementing reputation-based schemes [112], [113] that track the contributions of each participant encourages fair participation and tackles free riding attacks by rewarding participants who contribute with more resources to the training process.
- Data partitioning: Partitioning the training data [114], [115], [116] is found to be helpful in preventing malicious participants from accessing the whole data without providing enough resources during the training process.
- Token-based schemes: Implementing token-based schemes [117], or leveraging a blockchain network [118], [119], allows for ensuring that all participants contribute fairly to the training process by requiring participants to provide a certain number of tokens in exchange for access to the model or training data.
- Federated learning: It can also help in preventing the free riding attacks since it requires from all participants to train their local models with their own data and share the trained models with others. Hence, it ensures that all participants have contributed to the training process [120], [121], [122].

C. LESSONS LEARNED

Based on the conducted review of diverse privacy and security attacks and their defensive mechanisms in distributed learning, several key lessons can be gleaned, including:

1) Deploying appropriate security and privacy mechanisms is critical for protecting the integrity of the data, models, and distributed systems.
2) To minimize the impact of privacy attacks, it is essential to utilize suitable privacy-preserving measures and techniques, such as differential privacy, federated learning, secure multi-party computation, and homomorphic encryption. These techniques aid in safeguarding the confidentiality of both the data and the model while still enabling efficient distributed learning.
3) By leveraging a combination of the privacy and security mechanisms described above, the probability of encountering privacy and security attacks can be considerably diminished in distributed ML systems. Specifically, by implementing multiple layers of protection, such as access controls, data masking, secure communication protocols, and differential privacy, the security of the ML model and the associated data can be strengthened. However, it is crucial to select and implement the appropriate mechanisms based on the considered context and potential threats to ensure comprehensive protection against diverse attacks.
4) Setting up access controls, monitoring for unusual activities, and regularly updating security protocols can help prevent and identify the risks of attacks and vulnerabilities.
5) Collaboration among different participants should be maintained on trust and transparent environment to ensure the success of distributed learning.

VI. OPEN CHALLENGES AND FUTURE DIRECTIONS

In this section, we outline and briefly discuss crucial research challenges that require attention for future improvements in existing solutions and successful implementation of distributed learning systems (see Figure 8).

A. ARCHITECTURE SELECTION

In distributed learning, architecture selection involves choosing the hardware and software configuration for machines collaborating in ML model training. Key factors to consider include hardware, network architecture, and security risks. In architecture selection for distributed learning, hardware is a pivotal factor. The hardware chosen must balance power with cost-effectiveness, accommodating the workload. This involves selecting machines with suitable CPU, GPU, and memory resources to match model requirements. For instance, memory-intensive or parallelized models might necessitate machines with greater memory or more GPUs. The network architecture is also an important consideration in architecture selection for distributed learning. The network should be able to support the communication

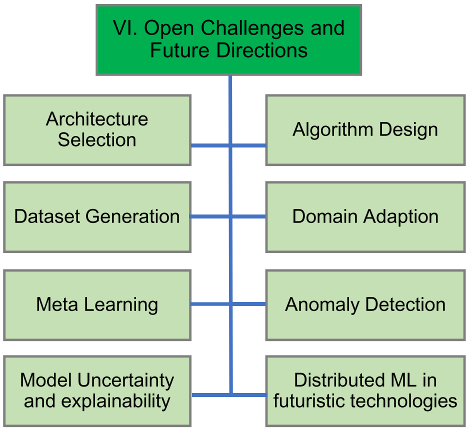

VI. Open Challenges and Future Directions
Architecture Selection
Algorithm Design
Dataset Generation
Domain Adaption
Meta Learning
Anomaly Detection
Model Uncertainty and explainability
Distributed ML in futuristic technologies

FIGURE 8. The proposed future research directions.

requirements of the distributed learning system. This may involve choosing high-bandwidth networking technologies, such as 5G networks, or designing a distributed system that can operate efficiently over a wide area network. Network architecture is also pivotal in distributed learning's architecture selection. The network must support system communication and address security concerns. This could involve high-bandwidth options or creating a distributed system efficient for wide area networks.

B. ALGORITHM DESIGN

It refers to the process of designing machine learning algorithms that are suitable for training on multiple machines. Several factors to consider when designing algorithms for distributed learning include data distribution, communication efficiency, scalability, convergence, and robustness. First, data is usually partitioned and distributed across multiple machines. Hence, learning algorithms must be tailored to efficiently process this data distribution. This can mean designing algorithms to work on local data subsets and aggregate results across machines, or employing distributed processing methods. Secondly, communication efficiency is crucial. Participating nodes must communicate for data and model parameter exchange. Thus, learning algorithms should minimize communication overheads to prevent learning slowdown. This could mean designing algorithms for distributed data processing, sending only essential information [123], or leveraging data sparsity for reduced communication overhead.

Scalability, rapid convergence, and Robustness are also pivotal. The collaborative learning algorithm should scale adeptly with machines and data size. This can entail designing algorithms for parallel data processing, employing model or data parallelism for workload distribution. The algorithm should swiftly converge to a satisfactory solution within resource constraints. This might involve designing algorithms that efficiently utilize available resources, incorporating techniques like stochastic gradient descent or mini-batch processing to enhance convergence speed. Moreover, ensuring chosen learning algorithms are resilient against communication failures, delays, or security/privacy risks is crucial in distributed contexts. This could involve designing algorithms to manage data inconsistencies or communication lags, or implementing fault tolerance and distributed consensus techniques to uphold training process integrity.

C. DATASET GENERATION

It is the process of creating a dataset that can be used to train a ML model on multiple machines. This is a critical step in distributed learning, as it involves collecting, preprocessing, and partitioning data to create a dataset that can be used to train a ML model in a distributed setting. There are several approaches to generating datasets for distributed learning, including data collection, data preprocessing, data augmentation, and data partitioning. Data collection involves collecting data from a diverse type of sources, such as sensors, databases, or the web, and storing it in a format that can be used to train a machine learning model. This can be a time-consuming and resource-intensive process, especially in distributed learning, as it may involve collecting large amounts of data from multiple sources and storing it on multiple machines. Data preprocessing involves cleaning and preparing the data for use in a ML model. This may consist of tasks such as removing missing or invalid data, standardizing the data, and splitting the data into training and testing sets. Data preprocessing is important in distributed learning, as it ensures that the data is in a consistent and usable format across all machines. Data augmentation composed of generating new data points based on existing data points in the dataset. This can be useful for increasing the size and diversity of the dataset and improving the performance of the machine learning model. In distributed learning, data augmentation can be used to generate additional data that can be distributed across multiple machines to improve the efficiency and scalability of the training process. Data partitioning involves dividing the dataset into smaller subsets, which can be stored and processed on different machines. This can improve the efficiency, efficacy and scalability of the training process, as it allows the model to be trained on smaller subsets of the data in parallel.

D. META-LEARNING

Meta-learning, or "learning to learn", is a subfield of ML that empowers learning models to enhance their performance on new tasks by leveraging past task experiences. It proves valuable in dynamic and diverse data and task environments, enabling models to swiftly adapt and elevate performance as conditions evolve. In distributed learning, meta-learning [124], [125] can be used to train models that are able to learn more efficiently and effectively using data distributed across multiple machines. Furthermore, meta-learning can also be leveraged to optimize the hyperparameters of ML models in a distributed setting. For example, a meta-learning algorithm could be used to learn the optimal learning rate or regularization parameters for a given task based on the performance of the model on previous tasks. Thus, meta-learning has the potential to significantly improve the efficiency and effectiveness of distributed learning by enabling ML models to adapt and improve their performance on new tasks using the knowledge gained from previous tasks.

E. DOMAIN ADAPTATION

Domain adaptation is a ML technique that involves adapting a model trained on one domain (or dataset) to a different, but related, domains. Domain adaptation is useful in situations where it is difficult or expensive to collect sufficient labeled data for a particular task, and instead, data from a related domain can be used to train a model that can then be adapted to the target domain [126]. In Particular, domain adaptation in distributed learning setup can be used to train models that can perform well on a variety of related tasks or domains using data distributed across multiple machines. This can be achieved through the use of techniques such as transfer learning, which allows a model to transfer knowledge learned on one domain to a related domain, and multi-task learning, which allows a model to learn multiple tasks or domains simultaneously. It is also possible to use domain adaptation in combination with meta-learning to further improve the performance of a learning model in a new domain. For example, a meta-learning algorithm could be used to optimize the hyperparameters of a model based on its performance on a variety of related tasks or domains.

F. ANOMALY DETECTION

Anomaly detection, is the process of identifying unusual or unexpected data points within a dataset. It is useful in a wide range of applications, including fraud detection, cybersecurity, and predictive maintenance. In distributed learning, anomaly detection can be challenging due to the large size and complexity of the data being analyzed. One approach to anomaly detection [127], [128] in distributed learning is to utilize unsupervised learning algorithms, such as clustering or density-based methods, which can determine patterns and anomalies in the data without the need for labeled examples. Another approach is to use supervised learning algorithms, which require a labeled training dataset to learn the typical behavior of the system being monitored. These algorithms can then be used to identify outliers from normal behavior as anomalies. It is also possible to use a combination of unsupervised and supervised learning techniques to improve the accuracy and reliability of the anomaly detection process. In summary, the process of identifying abnormal or unexpected data points in vast and intricate datasets within distributed learning requires the utilization of specific algorithms and methodologies.

G. MODEL UNCERTAINTY AND EXPLAINABILITY

Model uncertainty and explainability are significant considerations in distributed ML, as they can influence the reliability and trustworthiness of the results produced by the learning model. Model uncertainty refers to the degree of uncertainty or variability in the predictions made by a model. In distributed ML, model uncertainty can arise due to a variety of factors, including the variability in the data used to train the model, the complexity of the model [129], [130], and the number of parameters used in the model. To address model uncertainty, it is often useful to measure the uncertainty of the model's predictions using techniques such as Bayesian inference or bootstrapping.

Explainability, on the other hand, refers to the ability of a model to provide clear and interpretable explanations for its predictions. In distributed ML, it is important to ensure that the model is transparent and easy to understand, especially if the model is being used to make important decisions. In general, it is important to carefully consider model uncertainty and explainability in distributed ML to ensure that the model is reliable and trustworthy, and that its predictions can be understood and interpreted by human users. Approaches like feature importance analysis and model distillation can be good candidates to enhance the interpretability of ML models.

H. DISTRIBUTED ML IN FUTURISTIC TECHNOLOGIES

As we discussed that distributed ML focuses on training models on multiple machines, possibly with different hardware and software configurations, to improve the speed and scalability of the training process. Thus, it has the potential to be used in a wide range of futuristic technologies, including the following:

1) Autonomous vehicles: Distributed ML can be used to train models that enable autonomous vehicles [131], [132] to make decisions based on data collected from sensors and other sources.
2) Smart cities: Distributed ML [133], [134] can be used to interpret data from sensors and other sources in real-time to optimize resource allocation and improve the efficiency of city services.
3) Internet of Things (IoT): Real-time analysis of data from IoT devices can be facilitated by distributed learning, leading to the development of smarter and more responsive systems [135].
4) Personal assistants: Utilizing distributed learning, models can be trained to enhance the accuracy and efficiency of voice recognition in personal assistants like smart speakers, improving their ability to comprehend and respond to voice commands [63].
5) Healthcare: Distributed ML can be used to analyze medical data to spot patterns and trends that can inform treatment decisions and improve patient outcomes [136].

VII. CONCLUSION

The burgeoning technology of distributed learning can effectively harness the growing volume of data in distributed environments. However, the rise of this technology has raised additional concerns regarding the privacy and security of the users who are involved in the learning process. Thus, in this survey, we initially focused on examining the security and privacy challenges that arise with distributed machine learning. We subsequently conducted a comprehensive overview of various defensive mechanisms presented in existing literature, which aim to tackle these types of attacks. Furthermore, this survey identified the unique features of adversaries at various levels of the distributed learning systems, while also outlining several research challenges and potential directions for future investigations in this area. Therefore, this survey provides a valuable source of information for readers who wish to gain a more comprehensive insight into this topic, and encourages further exploration and advancement in the field of distributed machine learning.

REFERENCES

[1] T. Alam, "A reliable communication framework and its use in Internet of Things (IoT)," *CSEIT1835111*, vol. 10, pp. 450–456, May 2018.

[2] C. Briggs, Z. Fan, and P. Andras, "A review of privacy preserving federated learning for private IoT analytics," 2020, *arXiv:2004.11794*.

[3] A. I. Chen, M. L. Balter, T. J. Maguire, and M. L. Yarmush, "Deep learning robotic guidance for autonomous vascular access," *Nature Mach. Intell.*, vol. 2, no. 2, pp. 104–115, Feb. 2020.

[4] X. Bai, Z. Liang, Z. Zhu, A. Schwing, D. Forsyth, and V. Gruev, "Polarization-based underwater geolocalization with deep learning," *eLight*, vol. 3, no. 1, p. 15, Jul. 2023.

[5] W. Shi, Z. Huang, H. Huang, C. Hu, M. Chen, S. Yang, and H. Chen, "LOEN: Lensless opto-electronic neural network empowered machine vision," *Light: Sci. Appl.*, vol. 11, no. 1, p. 121, May 2022.

[6] C. K. Sahu, C. Young, and R. Rai, "Artificial intelligence (AI) in augmented reality (AR)-assisted manufacturing applications: A review," *Int. J. Prod. Res.*, vol. 59, no. 16, pp. 4903–4959, Aug. 2021.

[7] M. Satyanarayanan, "The emergence of edge computing," *Computer*, vol. 50, no. 1, pp. 30–39, Jan. 2017.

[8] Statista. (Sep. 2022). *Total Data Volume Worldwide 2010-2025*. [Online]. Available: https://www.statista.com/statistics/871513/worldwide-data-created/

[9] A. A. Abdellatif, A. Mohamed, C. F. Chiasserini, M. Tlili, and A. Erbad, "Edge computing for smart health: Context-aware approaches, opportunities, and challenges," *IEEE Netw.*, vol. 33, no. 3, pp. 196–203, May 2019.

[10] M. Usman, J. Rains, T. J. Cui, M. Z. Khan, J. U. R. Kazim, M. A. Imran, and Q. H. Abbasi, "Intelligent wireless walls for contactless in-home monitoring," *Light, Sci. Appl.*, vol. 11, no. 1, p. 212, Jul. 2022.

[11] A. A. Abdellatif, A. Mohamed, C. F. Chiasserini, A. Erbad, and M. Guizani, "Edge computing for energy-efficient smart health systems: Data and application-specific approaches," in *Energy Efficiency of Medical Devices and Healthcare Applications*. Amsterdam, The Netherlands: Elsevier, 2020, pp. 53–67.

[12] X. Cao, T. Basar, S. Diggavi, Y. C. Eldar, K. B. Letaief, H. V. Poor, and J. Zhang, "Communication-efficient distributed learning: An overview," *IEEE J. Sel. Areas Commun.*, vol. 41, no. 4, pp. 851–873, Apr. 2023.

[13] A. A. Abdellatif, C. F. Chiasserini, F. Malandrino, A. Mohamed, and A. Erbad, "Active learning with noisy labelers for improving classification accuracy of connected vehicles," *IEEE Trans. Veh. Technol.*, vol. 70, no. 4, pp. 3059–3070, Apr. 2021.

[14] Z. Ye, Y. J. Kumar, G. O. Sing, F. Song, and J. Wang, "A comprehensive survey of graph neural networks for knowledge graphs," *IEEE Access*, vol. 10, pp. 75729–75741, 2022.

[15] J. Verbraeken, M. Wolting, J. Katzy, J. Kloppenburg, T. Verbelen, and J. S. Rellermeyer, "A survey on distributed machine learning," *ACM Comput. Surveys*, vol. 53, no. 2, pp. 1–33, Mar. 2020, doi: 10.1145/3377454.

[16] H. C. Tanuwidjaja, R. Choi, S. Baek, and K. Kim, "Privacy-preserving deep learning on machine learning as a service—A comprehensive survey," *IEEE Access*, vol. 8, pp. 167425–167447, 2020.

[17] O. Nassef, W. Sun, H. Purmehdi, M. Tatipamula, and T. Mahmoodi, "A survey: Distributed machine learning for 5G and beyond," *Comput. Netw.*, vol. 207, Apr. 2022, Art. no. 108820. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1389128622000421

[18] E. Muscinelli, S. S. Shinde, and D. Tarchi, "Overview of distributed machine learning techniques for 6G networks," *Algorithms*, vol. 15, no. 6, p. 210, 2022. [Online]. Available: https://www.mdpi.com/1999-4893/15/6/210

[19] C. P. Filho, E. Marques, V. Chang, L. dos Santos, F. Bernardini, P. F. Pires, L. Ochi, and F. C. Delicato, "A systematic literature review on distributed machine learning in edge computing," *Sensors*, vol. 22, no. 7, p. 2665, Mar. 2022.

[20] E. Antwi-Boasiako, S. Zhou, Y. Liao, Q. Liu, Y. Wang, and K. Owusu-Agyemang, "Privacy preservation in distributed deep learning: A survey on distributed deep learning, privacy preservation techniques used and interesting research directions," *J. Inf. Secur. Appl.*, vol. 61, Sep. 2021, Art. no. 102949.

[21] G. Xu, H. Li, H. Ren, K. Yang, and R. H. Deng, "Data security issues in deep learning: Attacks, countermeasures, and opportunities," *IEEE Commun. Mag.*, vol. 57, no. 11, pp. 116–122, Nov. 2019.

[22] M. Gong, Y. Xie, K. Pan, K. Feng, and A. K. Qin, "A survey on differentially private machine learning," *IEEE Comput. Intell. Mag.*, vol. 15, no. 2, pp. 49–64, May 2020.

[23] D. Enthoven and Z. Al-Ars, "An overview of federated deep learning privacy attacks and defensive strategies," in *Federated Learning Systems: Towards Next-Generation AI*. 2021, pp. 173–196.

[24] L. Lyu, H. Yu, and Q. Yang, "Threats to federated learning: A survey," 2020, *arXiv:2003.02133*.

[25] J. Liu, J. Huang, Y. Zhou, X. Li, S. Ji, H. Xiong, and D. Dou, "From distributed machine learning to federated learning: A survey," *Knowl. Inf. Syst.*, vol. 64, no. 4, pp. 885–917, Apr. 2022.

[26] H. Li, C. Li, J. Wang, A. Yang, Z. Ma, Z. Zhang, and D. Hua, "Review on security of federated learning and its application in healthcare," *Future Gener. Comput. Syst.*, vol. 144, pp. 271–290, Jul. 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167739X23000626

[27] F. Zerka, S. Barakat, S. Walsh, M. Bogowicz, R. T. H. Leijenaar, A. Jochems, B. Miraglio, D. Townend, and P. Lambin, "Systematic review of privacy-preserving distributed machine learning from federated databases in health care," *JCO Clin. Cancer Informat.*, no. 4, pp. 184–200, Nov. 2020.

[28] J. Á. Morell and E. Alba, "Dynamic and adaptive fault-tolerant asynchronous federated learning using volunteer edge devices," *Future Gener. Comput. Syst.*, vol. 133, pp. 53–67, Aug. 2022.

[29] I. P. Egwutuoha, D. Levy, B. Selic, and S. Chen, "A survey of fault tolerance mechanisms and checkpoint/restart implementations for high performance computing systems," *J. Supercomput.*, vol. 65, no. 3, pp. 1302–1326, Sep. 2013.

[30] D. Das, S. Avancha, D. Mudigere, K. Vaidynathan, S. Sridharan, D. Kalamkar, B. Kaul, and P. Dubey, "Distributed deep learning using synchronous stochastic gradient descent," 2016, *arXiv:1602.06709*.

[31] D. Cheng, S. Li, H. Zhang, F. Xia, and Y. Zhang, "Why dataset properties bound the scalability of parallel machine learning training algorithms," *IEEE Trans. Parallel Distrib. Syst.*, vol. 32, no. 7, pp. 1702–1712, Jul. 2021.

[32] A. A. Abdellatif, N. Mhaisen, A. Mohamed, A. Erbad, M. Guizani, Z. Dawy, and W. Nasreddine, "Communication-efficient hierarchical federated learning for IoT heterogeneous systems with imbalanced data," *Future Gener. Comput. Syst.*, vol. 128, pp. 406–419, Mar. 2022.

[33] N. Mhaisen, A. A. Abdellatif, A. Mohamed, A. Erbad, and M. Guizani, "Optimal user-edge assignment in hierarchical federated learning based on statistical properties and network topology constraints," *IEEE Trans. Netw. Sci. Eng.*, vol. 9, no. 1, pp. 55–66, Jan. 2022.

[34] S. Ji, T. Saravirta, S. Pan, G. Long, and A. Walid, "Emerging trends in federated learning: From model fusion to federated X learning," 2021, *arXiv:2102.12920*.

[35] F. Sattler, S. Wiedemann, K.-R. Müller, and W. Samek, "Robust and communication-efficient federated learning from non-IID data," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 9, pp. 3400–3413, Sep. 2020.

[36] M. Khosravy, K. Nakamura, Y. Hirose, N. Nitta, and N. Babaguchi, "Model inversion attack by integration of deep generative models: Privacy-sensitive face generation from a face recognition system," *IEEE Trans. Inf. Forensics Security*, vol. 17, pp. 357–372, 2022.

[37] X. Yuan, L. Ding, L. Zhang, X. Li, and D. O. Wu, "ES attack: Model stealing against deep neural networks without data hurdles," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 6, no. 5, pp. 1258–1270, Oct. 2022.

[38] S. Truex, L. Liu, M. E. Gursoy, L. Yu, and W. Wei, "Demystifying membership inference attacks in machine learning as a service," *IEEE Trans. Services Comput.*, vol. 14, no. 6, pp. 2073–2089, Nov. 2021.

[39] C. J. D'Orazio, K. R. Choo, and L. T. Yang, "Data exfiltration from Internet of Things devices: IOS devices as case studies," *IEEE Internet Things J.*, vol. 4, no. 2, pp. 524–535, Apr. 2017.

[40] J. Zhang, B. Chen, X. Cheng, H. T. T. Binh, and S. Yu, "PoisonGAN: Generative poisoning attacks against federated learning in edge computing systems," *IEEE Internet Things J.*, vol. 8, no. 5, pp. 3310–3322, Mar. 2021.

[41] S. Zhao, X. Ma, X. Zheng, J. Bailey, J. Chen, and Y.-G. Jiang, "Clean-label backdoor attacks on video recognition models," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 14431–14440.

[42] M. Tajmirriahi, R. Kafieh, Z. Amini, and V. Lakshminarayanan, "A dual-discriminator Fourier acquisitive GAN for generating retinal optical coherence tomography images," *IEEE Trans. Instrum. Meas.*, vol. 71, pp. 1–8, 2022.

[43] H. Zhu, R. Leung, and M. Hong, "Shadow compensation for synthetic aperture radar target classification by dual parallel generative adversarial network," *IEEE Sensors Lett.*, vol. 4, no. 8, pp. 1–4, Aug. 2020.

[44] S.-H. Choi, J.-M. Shin, P. Liu, and Y.-H. Choi, "ARGAN: Adversarially robust generative adversarial networks for deep neural networks against adversarial examples," *IEEE Access*, vol. 10, pp. 33602–33615, 2022.

[45] G. Shao, M. Huang, F. Gao, T. Liu, and L. Li, "DuCaGAN: Unified dual capsule generative adversarial network for unsupervised image-to-image translation," *IEEE Access*, vol. 8, pp. 154691–154707, 2020.

[46] D. Wang, L. Dong, R. Wang, D. Yan, and J. Wang, "Targeted speech adversarial example generation with generative adversarial network," *IEEE Access*, vol. 8, pp. 124503–124513, 2020.

[47] A. Bhardwaj, V. Mangat, and R. Vig, "Hyperband tuned deep neural network with well posed stacked sparse AutoEncoder for detection of DDoS attacks in cloud," *IEEE Access*, vol. 8, pp. 181916–181929, 2020.

[48] Z. Zhang, R. Deng, D. K. Y. Yau, and P. Chen, "Zero-parameter-information data integrity attacks and countermeasures in IoT-based smart grid," *IEEE Internet Things J.*, vol. 8, no. 8, pp. 6608–6623, Apr. 2021.

[49] Y. Yu, L. Guo, Y. Liu, J. Zheng, and Y. Zong, "An efficient SDN-based DDoS attack detection and rapid response platform in vehicular networks," *IEEE Access*, vol. 6, pp. 44570–44579, 2018.

[50] J. C. Palencia, M. G. Harbour, J. J. Gutiérrez, and J. M. Rivas, "Response-time analysis in hierarchically-scheduled time-partitioned distributed systems," *IEEE Trans. Parallel Distrib. Syst.*, vol. 28, no. 7, pp. 2017–2030, Jul. 2017.

[51] V. Rao and K. V. Prema, "Performing real-time network attacks on smart weather monitoring device using kali Linux," in *Proc. IEEE Bengaluru Humanitarian Technol. Conf. (B-HTC)*, Oct. 2020, pp. 1–6.

[52] M. M. Khalili, X. Zhang, and M. Liu, "Designing contracts for trading private and heterogeneous data using a biased differentially private algorithm," *IEEE Access*, vol. 9, pp. 70732–70745, 2021.

[53] F. Shang, T. Xu, Y. Liu, H. Liu, L. Shen, and M. Gong, "Differentially private ADMM algorithms for machine learning," *IEEE Trans. Inf. Forensics Security*, vol. 16, pp. 4733–4745, 2021.

[54] M.-F. Balcan, T. Dick, and E. Vitercik, "Dispersion for data-driven algorithm design, online learning, and private optimization," in *Proc. IEEE 59th Annu. Symp. Found. Comput. Sci. (FOCS)*, Oct. 2018, pp. 603–614.

[55] G. Qiu, X. Gui, and Y. Zhao, "Privacy-preserving linear regression on distributed data by homomorphic encryption and data masking," *IEEE Access*, vol. 8, pp. 107601–107613, 2020.

[56] P. Chevalier, P. Quéméré, S. Bérard-Bergery, J.-B. Henry, C. Beylier, and J. Vaillant, "Rigorous model-based mask data preparation algorithm applied to grayscale lithography for the patterning at the micrometer scale," *J. Microelectromech. Syst.*, vol. 30, no. 3, pp. 442–455, Jun. 2021.

[57] X. Yue, J. Lin, F. R. Gutierrez, and H. Li, "Self-supervised learning with segmental masking for speech representation," *IEEE J. Sel. Topics Signal Process.*, vol. 16, no. 6, pp. 1367–1379, Oct. 2022.

[58] P. Li, J. Su, and X. Wang, "ITLS: Lightweight transport-layer security protocol for IoT with minimal latency and perfect forward secrecy," *IEEE Internet Things J.*, vol. 7, no. 8, pp. 6828–6841, Aug. 2020.

[59] E. B. Sanjuan, I. A. Cardiel, J. A. Cerrada, and C. Cerrada, "Message queuing telemetry transport (MQTT) security: A cryptographic smart card approach," *IEEE Access*, vol. 8, pp. 115051–115062, 2020.

[60] A. F. De Abiega-L'Eglisse, K. A. Delgado-Vargas, F. Q. Valencia-Rodriguez, V. G. Gonzalez-Quiroga, G. Gallegos-Garcia, and M. Nakano-Miyatake, "Performance of new hope and CRYSTALS-dilithium postquantum schemes in the transport layer security protocol," *IEEE Access*, vol. 8, pp. 213968–213980, 2020.

[61] A. Liu, A. Alqazzaz, H. Ming, and B. Dharmalingam, "Iotverif: Automatic verification of SSL/TLS certificate for IoT applications," *IEEE Access*, vol. 9, pp. 27038–27050, 2021.

[62] M. Uddin, S. Islam, and A. Al-Nemrat, "A dynamic access control model using authorising workflow and task-role-based access control," *IEEE Access*, vol. 7, pp. 166676–166689, 2019.

[63] M. Shen, X. Tang, L. Zhu, X. Du, and M. Guizani, "Privacy-preserving support vector machine training over blockchain-based encrypted IoT data in smart cities," *IEEE Internet Things J.*, vol. 6, no. 5, pp. 7702–7712, Oct. 2019.

[64] A. Pastor, A. Mozo, S. Vakaruk, D. Canavese, D. R. Lopez, L. Regano, S. Gomez-Canaval, and A. Lioy, "Detection of encrypted cryptomining malware connections with machine and deep learning," *IEEE Access*, vol. 8, pp. 158036–158055, 2020.

[65] H. Wu, G. Liu, Y. Yao, and X. Zhang, "Watermarking neural networks with watermarked images," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 7, pp. 2591–2601, Jul. 2021.

[66] S. Rani and R. Halder, "Comparative analysis of relational database watermarking techniques: An empirical study," *IEEE Access*, vol. 10, pp. 27970–27989, 2022.

[67] C.-L. Liu and P.-Y. Hsieh, "Model-based synthetic sampling for imbalanced data," *IEEE Trans. Knowl. Data Eng.*, vol. 32, no. 8, pp. 1543–1556, Aug. 2020.

[68] F. K. Dankar, M. K. Ibrahim, and L. Ismail, "A multi-dimensional evaluation of synthetic data generators," *IEEE Access*, vol. 10, pp. 11147–11158, 2022.

[69] J. Liu, F. Qu, X. Hong, and H. Zhang, "A small-sample wind turbine fault detection method with synthetic fault data using generative adversarial nets," *IEEE Trans. Ind. Informat.*, vol. 15, no. 7, pp. 3877–3888, Jul. 2019.

[70] E. Bou-Harb, M. Husák, M. Debbabi, and C. Assi, "Big data sanitization and cyber situational awareness: A network telescope perspective," *IEEE Trans. Big Data*, vol. 5, no. 4, pp. 439–453, Dec. 2019.

[71] J. C. Lin, J. M. Wu, P. Fournier-Viger, Y. Djenouri, C.-H. Chen, and Y. Zhang, "A sanitization approach to secure shared data in an IoT environment," *IEEE Access*, vol. 7, pp. 25359–25368, 2019.

[72] J. Cui, W. Liu, J. Huang, and L. T. Yang, "ADS: Leveraging approximate data for efficient data sanitization in SSDs," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 41, no. 6, pp. 1771–1784, Jun. 2022.

[73] A. Majeed and S. Lee, "Anonymization techniques for privacy preserving data publishing: A comprehensive survey," *IEEE Access*, vol. 9, pp. 8512–8545, 2021.

[74] M. Yuan, L. Chen, P. S. Yu, and T. Yu, "Protecting sensitive labels in social network data anonymization," *IEEE Trans. Knowl. Data Eng.*, vol. 25, no. 3, pp. 633–647, Mar. 2013.

[75] J. Feng, L. T. Yang, Q. Zhu, and K. R. Choo, "Privacy-preserving tensor decomposition over encrypted data in a federated cloud environment," *IEEE Trans. Dependable Secure Comput.*, vol. 17, no. 4, pp. 857–868, Jul. 2020.

[76] Q. Li, Y. Ju, C. Zhao, and X. He, "An encrypted field locating algorithm for private protocol data based on data reconstruction and moment eigenvector," *IEEE Access*, vol. 9, pp. 42947–42958, 2021.

[77] R. Hu, M. Ye, C. Ma, and F. Chen, "Distributed supervised discrete hashing with relaxation," *IEEE Access*, vol. 9, pp. 63729–63739, 2021.

[78] M. Thangavel and P. Varalakshmi, "Enabling ternary hash tree based integrity verification for secure cloud data storage," *IEEE Trans. Knowl. Data Eng.*, vol. 32, no. 12, pp. 2351–2362, Dec. 2020.

[79] J. Chen, X. Zhang, R. Zhang, C. Wang, and L. Liu, "De-pois: An attack-agnostic defense against data poisoning attacks," *IEEE Trans. Inf. Forensics Security*, vol. 16, pp. 3412–3425, 2021.

[80] R. Catelli, F. Gargiulo, V. Casola, G. De Pietro, H. Fujita, and M. Esposito, "A novel COVID-19 data set and an effective deep learning approach for the de-identification of Italian medical records," *IEEE Access*, vol. 9, pp. 19097–19110, 2021.

[81] J. Li, R. C.-W. Wong, A. Wai-Chee Fu, and J. Pei, "Anonymization by local recoding in data with attribute hierarchical taxonomies," *IEEE Trans. Knowl. Data Eng.*, vol. 20, no. 9, pp. 1181–1194, Sep. 2008.

[82] X. Zhang, Z. Wang, D. Liu, Q. Lin, and Q. Ling, "Deep adversarial data augmentation for extremely low data regimes," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 1, pp. 15–28, Jan. 2021.

[83] K. Seo, H. Cho, D. Choi, and J.-D. Park, "Implicit semantic data augmentation for hand pose estimation," *IEEE Access*, vol. 10, pp. 84680–84688, 2022.

[84] D. Liu, L. Zhang, X. Jiang, C. Su, Y. Fan, and Y. Cao, "MEML: A deep data augmentation method by mean extrapolation in middle layers," *IEEE Access*, vol. 9, pp. 151621–151630, 2021.

[85] Q. Liu, J. Guo, C.-K. Wen, and S. Jin, "Adversarial attack on DL-based massive MIMO CSI feedback," *J. Commun. Netw.*, vol. 22, no. 3, pp. 230–235, Jun. 2020.

[86] H. Mun, S. Seo, B. Son, and J. Yun, "Black-box audio adversarial attack using particle swarm optimization," *IEEE Access*, vol. 10, pp. 23532–23544, 2022.

[87] S. Chaudhury, H. Roy, S. Mishra, and T. Yamasaki, "Adversarial training time attack against discriminative and generative convolutional models," *IEEE Access*, vol. 9, pp. 109241–109259, 2021.

[88] I. Alsmadi, N. Aljaafari, M. Nazzal, S. Alhamed, A. H. Sawalmeh, C. P. Vizcarra, A. Khreishah, M. Anan, A. Algosaibi, M. A. Al-Naeem, A. Aldalbahi, and A. Al-Humam, "Adversarial machine learning in text processing: A literature survey," *IEEE Access*, vol. 10, pp. 17043–17077, 2022.

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

[90] Y. Emre Isik, Y. Gormez, O. Kaynar, and Z. Aydin, "NSEM: Novel stacked ensemble method for sentiment analysis," in *Proc. Int. Conf. Artif. Intell. Data Process. (IDAP)*, Sep. 2018, pp. 1–4.

[91] V. Ivanyuk, A. Tsvirkun, V. Soloviev, V. Feklin, A. Sunchalina, and O. Kravchenko, "Forecasting financial time series based on the ensemble method," in *Proc. 15th Int. Conf. Manage. Large-Scale Syst. Develop. (MLSD)*, Sep. 2022, pp. 1–5.

[92] Z. Zhu, Z. Wang, D. Li, Y. Zhu, and W. Du, "Geometric structural ensemble learning for imbalanced problems," *IEEE Trans. Cybern.*, vol. 50, no. 4, pp. 1617–1629, Apr. 2020.

[93] L. C. F. Domingos, P. E. Santos, P. S. M. Skelton, R. S. A. Brinkworth, and K. Sammut, "An investigation of preprocessing filters and deep learning methods for vessel type classification with underwater acoustic data," *IEEE Access*, vol. 10, pp. 117582–117596, 2022.

[94] K. Xu, Y. Zhong, and G. Wang, "A hybrid regularization technique for solving highly nonlinear inverse scattering problems," *IEEE Trans. Microw. Theory Techn.*, vol. 66, no. 1, pp. 11–21, Jan. 2018.

[95] T. Kim and S.-Y. Yun, "Revisiting orthogonality regularization: A study for convolutional neural networks in image classification," *IEEE Access*, vol. 10, pp. 69741–69749, 2022.

[96] A. Abbas, M. M. Abdelsamea, and M. M. Gaber, "4S-DT: Self-supervised super sample decomposition for transfer learning with application to COVID-19 detection," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 7, pp. 2798–2808, Jul. 2021.

[97] S. A. H. Minoofam, A. Bastanfard, and M. R. Keyvanpour, "TRCLA: A transfer learning approach to reduce negative transfer for cellular learning automata," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 5, pp. 2480–2489, May 2023.

[98] Z. Sha, H. Shu, X. Xiong, and F. Kang, "Model of execution trace obfuscation between threads," *IEEE Trans. Dependable Secure Comput.*, vol. 19, no. 6, pp. 4156–4171, Nov. 2022.

[99] S. Afshari, M. Musisi-Nkambwe, and I. Sanchez Esqueda, "Analyzing the impact of memristor variability on crossbar implementation of regression algorithms with smart weight update pulsing techniques," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 69, no. 5, pp. 2025–2034, May 2022.

[100] Z.-B. Yu and M.-L. Zhang, "Multi-label classification with label-specific feature generation: A wrapped approach," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 9, pp. 5199–5210, Sep. 2022.

[101] B. E. ElDiwany, A. A. Abdellatif, A. Mohamed, A. Al-Ali, M. Guizani, and X. Du, "On physical layer security in energy-efficient wireless health monitoring applications," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2019, pp. 1–7.

[102] P. G. Shynu, H. Md. Shayan., and C. L. Chowdhary, "A fuzzy based data perturbation technique for privacy preserved data mining," in *Proc. Int. Conf. Emerg. Trends Inf. Technol. Eng. (ic-ETITE)*, Feb. 2020, pp. 1–4.

[103] S. Y. Chang and H.-C. Wu, "Multi-relational data characterization by tensors: Perturbation analysis," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 1, pp. 756–769, Jan. 2023.

[104] V. Cherkassky, X. Shao, F. M. Mulier, and V. N. Vapnik, "Model complexity control for regression using VC generalization bounds," *IEEE Trans. Neural Netw.*, vol. 10, no. 5, pp. 1075–1089, Jun. 1999.

[105] Y.-W. Hong, J.-S. Leu, M. Faisal, and S. W. Prakosa, "Analysis of model compression using knowledge distillation," *IEEE Access*, vol. 10, pp. 85095–85105, 2022.

[106] M. Rahimpour, J. Bertels, A. Radwan, H. Vandermeulen, S. Sunaert, D. Vandermeulen, F. Maes, K. Goffin, and M. Koole, "Cross-modal distillation to improve MRI-based brain tumor segmentation with missing MRI sequences," *IEEE Trans. Biomed. Eng.*, vol. 69, no. 7, pp. 2153–2164, Jul. 2022.

[107] B. Zhao, Q. Wang, Y. Wu, Q. Cao, and Q. Ran, "Target detection model distillation using feature transition and label registration for remote sensing imagery," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 5416–5426, 2022.

[108] J. Song, Y. Chen, J. Ye, and M. Song, "Spot-adaptive knowledge distillation," *IEEE Trans. Image Process.*, vol. 31, pp. 3359–3370, 2022.

[109] Y. Fan, J. Zhang, N. Zhao, Y. Ren, J. Wan, L. Zhou, Z. Shen, J. Wang, J. Zhang, and Z. Wei, "Model aggregation method for data parallelism in distributed real-time machine learning of smart sensing equipment," *IEEE Access*, vol. 7, pp. 172065–172073, 2019.

[110] N. Thorwirth, "The decentralized rights locker," *SMPTE Motion Imag. J.*, vol. 129, no. 3, pp. 56–62, Apr. 2020.

[111] J. Gao, H. Yu, X. Zhu, and X. Li, "Blockchain-based digital rights management scheme via multiauthority ciphertext-policy attribute-based encryption and proxy re-encryption," *IEEE Syst. J.*, vol. 15, no. 4, pp. 5233–5244, Dec. 2021.

[112] C. Huang, Z. Wang, H. Chen, Q. Hu, Q. Zhang, W. Wang, and X. Guan, "RepChain: A reputation-based secure, fast, and high incentive blockchain system via sharding," *IEEE Internet Things J.*, vol. 8, no. 6, pp. 4291–4304, Mar. 2021.

[113] H.-T. Wu, Y. Zheng, B. Zhao, and J. Hu, "An anonymous reputation management system for mobile crowdsensing based on dual blockchain," *IEEE Internet Things J.*, vol. 9, no. 9, pp. 6956–6968, May 2022.

[114] M. S. Mahmud, J. Z. Huang, S. Salloum, T. Z. Emara, and K. Sadatdiynov, "A survey of data partitioning and sampling methods to support big data analysis," *Big Data Mining Analytics*, vol. 3, no. 2, pp. 85–101, Jun. 2020.

[115] T. Z. Emara and J. Z. Huang, "Distributed data strategies to support large-scale data analysis across geo-distributed data centers," *IEEE Access*, vol. 8, pp. 178526–178538, 2020.

[116] S. Salloum, J. Z. Huang, and Y. He, "Random sample partition: A distributed data model for big data analysis," *IEEE Trans. Ind. Informat.*, vol. 15, no. 11, pp. 5846–5854, Nov. 2019.

[117] G. Gan, E. Chen, Z. Zhou, and Y. Zhu, "Token-based access control," *IEEE Access*, vol. 8, pp. 54189–54199, 2020.

[118] A. A. Abdellatif, A. Z. Al-Marridi, A. Mohamed, A. Erbad, C. F. Chiasserini, and A. Refaey, "SsHealth: Toward secure, blockchain-enabled healthcare systems," *IEEE Netw.*, vol. 34, no. 4, pp. 312–319, Jul. 2020.

[119] S. Shalaby, A. A. Abdellatif, A. Al-Ali, A. Mohamed, A. Erbad, and M. Guizani, "Performance evaluation of hyperledger fabric," in *Proc. IEEE Int. Conf. Informat., IoT, Enabling Technol. (ICIoT)*, Feb. 2020, pp. 608–613.

[120] B. Ghimire and D. B. Rawat, "Recent advances on federated learning for cybersecurity and cybersecurity for federated learning for Internet of Things," *IEEE Internet Things J.*, vol. 9, no. 11, pp. 8229–8249, Jun. 2022.

[121] B. Gu, A. Xu, Z. Huo, C. Deng, and H. Huang, "Privacy-preserving asynchronous vertical federated learning algorithms for multiparty collaborative learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 11, pp. 6103–6115, Nov. 2022.

[122] M. H. Shullary, A. A. Abdellatif, and Y. Massoudn, "Energy-efficient active federated learning on non-IID data," in *Proc. IEEE 65th Int. Midwest Symp. Circuits Syst. (MWSCAS)*, Aug. 2022, pp. 1–4.

[123] A. Abdellatif, "Novel processing and transmission techniques leveraging edge computing for smart health systems," Tech. Rep., 2018.

[124] J. Zhang, J. Song, L. Gao, Y. Liu, and H. T. Shen, "Progressive meta-learning with curriculum," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 9, pp. 5916–5930, Sep. 2022.

[125] Z. Li, J. Li, C. Wang, Z. Lu, J. Wang, H. He, and J. Shi, "Meta-learning based interactively connected clique U-Net for quantitative susceptibility mapping," *IEEE Trans. Comput. Imag.*, vol. 7, pp. 1385–1399, 2021.

[126] A. A. Abdellatif, C. F. Chiasserini, and F. Malandrino, "Active learning-based classification in automated connected vehicles," in *Proc. IEEE Conf. Comput. Commun. Workshops (INFOCOM WKSHPS)*, Jul. 2020, pp. 598–603.

[127] J. Luo, J. Lin, Z. Yang, and H. Liu, "SMD anomaly detection: A self-supervised texture–structure anomaly detection framework," *IEEE Trans. Instrum. Meas.*, vol. 71, 2022, Art. no. 5017611.

[128] A. Lundström, M. O'nils, F. Z. Qureshi, and A. Jantsch, "Improving deep learning based anomaly detection on multivariate time series through separated anomaly scoring," *IEEE Access*, vol. 10, pp. 108194–108204, 2022.

[129] A. Jung and P. H. J. Nardelli, "An information-theoretic approach to personalized explainable machine learning," *IEEE Signal Process. Lett.*, vol. 27, pp. 825–829, 2020.

[130] G. Pavlin, J. P. de Villiers, J. Ziegler, A.-L. Jousselme, P. Costa, K. Laskey, A. de Waal, E. Blasch, and L. Jansen, "Relations between explainability, evaluation and trust in AI-based information fusion systems," in *Proc. IEEE 24th Int. Conf. Inf. Fusion (FUSION)*, Jul. 2021, pp. 1–9.

[131] Y. Fu, C. Li, F. R. Yu, T. H. Luan, and Y. Zhang, "An autonomous lane-changing system with knowledge accumulation and transfer assisted by vehicular blockchain," *IEEE Internet Things J.*, vol. 7, no. 11, pp. 11123–11136, Nov. 2020.

[132] Y. He, K. Huang, G. Zhang, F. R. Yu, J. Chen, and J. Li, "Bift: A blockchain-based federated learning system for connected and autonomous vehicles," *IEEE Internet Things J.*, vol. 9, no. 14, pp. 12311–12322, Jul. 2022.

[133] L. Basheer and P. Ranjana, "A comparative study of various intrusion detections in smart cities using machine learning," in *Proc. Int. Conf. IoT Blockchain Technol. (ICIBT)*, May 2022, pp. 1–6.

[134] A. Sinaeepourfard, J. Krogstie, and S. Sengupta, "Distributed-to-centralized data management: A new sense of large-scale ICT management of smart city IoT networks," *IEEE Internet Things Mag.*, vol. 3, no. 3, pp. 76–82, Sep. 2020.

[135] R. Rodulfo, "Smart city case study: City of coral gables leverages the Internet of Things to improve quality of life," *IEEE Internet Things Mag.*, vol. 3, no. 2, pp. 74–81, Jun. 2020.

[136] G. Bansal, K. Rajgopal, V. Chamola, Z. Xiong, and D. Niyato, "Healthcare in metaverse: A survey on current metaverse applications in healthcare," *IEEE Access*, vol. 10, pp. 119914–119946, 2022.

MUHAMMAD USMAN AFZAL (Member, IEEE) received the B.S. degree (Hons.) in electrical engineering from the Information Technology University of the Punjab (ITU-Lahore), Lahore, Pakistan, in 2022. He is currently a Research Associate with the MicroNano Laboratory, ITU-Lahore. In 2021, during his senior year of undergraduate studies, he secured funding from the Undergraduate Research Outreach Program (UROP) for his final year project titled "Antenna Radiation Pattern Mapper Using Software Defined Radio (SDR)." Additionally, he actively contributes as a member of the IEEE Student Chapter at ITU-Lahore. His research interests include exploring the privacy and security aspects of distributed machine learning, AI-based bio-meta-photonics systems, and leveraging resource optimization techniques for power-efficient and secure FL procedures in UAVs. He is eager to make meaningful contributions to the field, by advancing the understanding and implementation of complex and collaborative scientific applications in the realms of distributed systems and photonics.

ALAA AWAD ABDELLATIF (Member, IEEE) received the B.Sc. and M.Sc. degrees (Hons.) in electronics and electrical communications engineering from Cairo University, in 2009 and 2012, respectively, and the Ph.D. degree from Politecnico di Torino, in 2018. He was a Research Assistant with Cairo University, from 2009 to 2012. He was a Senior Research Assistant with Qatar University, from 2013 to 2015. He was a Postdoctoral Researcher with Politecnico di Torino, from 2019 to 2020. He is currently a Postdoctoral Researcher with Qatar University. He has authored or coauthored over 52 refereed journal articles, magazine, and conference papers in reputable international journals and conferences. His research interests include edge computing, blockchain, machine learning, and resource optimization for wireless heterogeneous networks, smart health, the IoT applications, and vehicular networks. He was a recipient of the Postdoctoral Research Award and the Graduate Student Research Award from the Qatar National Research Fund, the Best Paper Award from the Wireless Telecommunications Symposium 2018, USA, in addition to the Quality Award from Politecnico di Torino, in 2018. He has served as a technical reviewer for many international journals and magazines.

MUHAMMAD ZUBAIR (Senior Member, IEEE) received the Ph.D. degree in computational electromagnetics (CEM) from the Polytechnic University of Turin, Italy. His Ph.D. research was carried out at the Antenna and EMC Laboratory (LACE), Istituto Superiore Mario Boella (ISMB), Turin, Italy, where he worked on computational EM projects funded by the Army Research Laboratory (ARL) in collaboration with Boston University, USA. He is currently associated with Innovative Technologies Laboratories, King Abdullah University of Science and Technology (KAUST), as a Research Scientist. Before joining KAUST, he was an Associate Professor and the Department Chair with the Department of Electrical Engineering, Information Technology University (ITU), Lahore. He was a Postdoctoral Research Fellow with the SUTD-MIT International Design Centre, Singapore, before joining ITU. During the postdoctoral research, he has been working on various research projects related to fractional methods in electromagnetics funded by the Air Force Office of Scientific Research (AFOSR)/Asian Office of Aerospace Research and Development (AOARD) and Singapore Temasek Laboratories (TL). He is the principal author of his book *Electromagnetic Fields and Waves in Fractional Dimensional Space* (Springer, NY). He has contributed over 50 scientific works in journals and conferences of international repute. He has been selected for the URSI Young Scientist Award (YSA) in the URSI General Assembly (GASS), in 2021. He received the Punjab Innovation Research Challenge Award (PIRCA), in 2021. He has been an Associate Editor of IEEE Access and an Editorial Board Member of the *IET Microwaves, Antennas & Propagation*, *PLOS One*, and *International Journal of Antennas and Propagation*. He continues to serve the Scientific Community as an Invited Book Reviewer for *Contemporary Physics* (Taylor & Francis).

MUHAMMAD QASIM MEHMOOD (Senior Member, IEEE) received the Ph.D. degree from the National University of Singapore, in 2016. He is currently an Associate Professor with the Department of Electrical Engineering, Information Technology University (ITU) of the Punjab, Lahore, Pakistan. He is also the Director of the Micro-Nano Laboratory (http://micronano.itu.edu.pk/), ITU. He secured several grants from various national and international funding agencies. He supervised/co-supervised several undergraduate and graduate theses. His research interests include metaoptics, metaphotonics, optical, photonics engineering, antenna microwave engineering, and printed electronics. He is a member of the Pak-ICTP Alumni Society. He was a recipient of the 2023 ICO/ICTP Gallieno Denardo Award for his remarkable contributions to the field of nano-optics and meta-photonics; and for inspiring many young researchers in Pakistan through quality research, mentorship, community services, and outreach activities. He is the Chair of the IEEE APS/CAS/MTT/SSC Joint Chapter of the Lahore Section (Region 10) and an Advisor of ITU's SPIE Chapter and OPTICA Chapter. He has been listed among the top 2% of scientists in a global list of 2021 and 2022 released by Stanford University. He works closely with other national and international societies to promote science by organizing various student development and outreach programs like symposiums, seminars, internships, project competitions, and awareness sessions. The prime focus is to target the unprivileged sections of society in Pakistan's far-reach low resources to promote science and encourage/expose students of all genders to science education. His efforts are helping immensely in advancing science and producing internationally competitive researchers from Pakistan.

YEHIA MASSOUD (Fellow, IEEE) is currently the Director of Innovative Technologies Laboratories (ITL), King Abdullah University of Science and Technology (KAUST). His research group was responsible for developing the world's first realization of compressive sensing systems for signals, which provided an unprecedented one-order of magnitude savings in power consumption and significant reductions in size and cost and has enabled the implementation of self-powered sensors for smart cities and ultra-low-power biomedical implantable devices. He has been a PI or a Co-PI on more than U.S. $30 Million of funded research from the NSF, DOD, SRC, and the industry. He has published more than 400 papers in leading peer-reviewed journals and conference publications. His research interests include the design of state-of-the-art innovative technological solutions that span a broad range of technical areas including smart cities, autonomy, smart health, smart mobility, embedded systems, nanophotonics, and spintronics. He was selected as one of ten MIT Alumni featured by MIT's Electrical Engineering and Computer Science Department, in 2012. He was a recipient of the Rising Star of Texas Medal, the National Science Foundation CAREER Award, the DAC Fellowship, the Synopsys Special Recognition Engineering Award, and several best paper awards. He also served as the 2016 IEEE MWSCAS Technical Program Co-Chair, the 2009 General Program Co-Chair, and the 2007 Technical Program Co-Chair of the ACM Great Lakes Symposium on VLSI. He has served as the Editor for the *Mixed-Signal Letters—The Americas*, an Associate Editor for IEEE Transactions on Very Large Scale Integration (VLSI) Systems and IEEE Transactions on Circuits and Systems—I: Regular Papers, and the Guest Editor for a Special Issue of IEEE Transactions on Circuits and Systems—I: Regular Papers. He has served on the IEEE CAS Award Nomination Committee, the IEEE Mac Valkenburg Award Committee, the IEEE CAS Fellow Committee, the IEEE Rebooting Computing Steering Committee, and the IEEE Nanotechnology Council. He was also named a Distinguished Lecturer by the IEEE Circuits and Systems Society.