Artificial Intelligence Applications And Innovations Vol. 1 No. 2 (2024)

Article history:
Received 3 January 2024
Revised 1 March 2024
Accepted 7 March 2024
Published online 02 April 2024

A Survey on Data Distribution Challenges and Solutions in Vertical and Horizontal Federated Learning

Khalil Jahani[1*], Behzad Moshiri[2], Babak Hossein Khalaj [3]

[1.]Department of Computer science, kish International Campus, University of Tehran, Tehran, Iran.
[2] Department of Electrical and Computer Engineering, College of Engineering, University of Tehran, Tehran, Iran.
[3.] Department of Electrical Engineering, Sharif University of Technology, Tehran, Iran.

*** Corresponding author email address**: jahanii@ut.ac.ir

A r t i c l e I n f o

Article type:
Original Research

How to cite this article:
Jahani, K., Moshiri, B., & Khalaj, B.H. (2024). A Survey on Data Distribution Challenges and Solutions in Vertical and Horizontal Federated Learning. Artificial Intelligence Applications and Innovations, *1*(2), 55-71
https://doi.org/10.61838/jaiai.1.2.5

A B S T R A C T

Federated learning is a novel way of training machine learning models on data that is distributed across multiple devices, such as smartphones and IoT sensors, without compromising privacy, efficiency, or security. However, federated learning faces a significant challenge when the data on each device is not independent and identically distributed (non-IID), which means that the data may have different distributions, sizes, or qualities. non-IID data is a major challenge for federated learning, as it affects the accuracy and participation of the local devices. Most existing methods focus on improving the model, algorithm, or framework of federated learning to deal with non-IID data. However, there is a lack of systematic and up-to-date reviews on this topic. In this paper, we survey different approaches to address the challenge of non-IID data in Vertical Federated Learning (VFL) and Horizontal Federated Learning (HFL). We organize the existing literature based on the perspective of the researcher and the sub-tasks involved in each approach. Our goal is to provide a comprehensive and systematic overview of the problem and its solutions.

Keywords: *non-IID data, Vertical Federated Learning, Horizontal Federated Learning.*

1. Introduction

Federated Learning (FL) represents a new paradigm in machine learning, enabling distributed entities to collaboratively train a global model while retaining their data locally. Unlike traditional centralized learning approaches, FL ensures data remains on individual devices, with only model updates, such as gradients, shared with a central aggregator. This decentralized framework significantly enhances data privacy and security, mitigating risks associated with data breaches and privacy violations. FL encompasses various data-sharing arrangements, as illustrated in Figure 1: Horizontal FL, where participants share similar feature types but distinct data instances; Vertical FL, where participants possess different features for the same data instances; and Federated Transfer Learning, which facilitates collaboration across differing data instances and feature types.

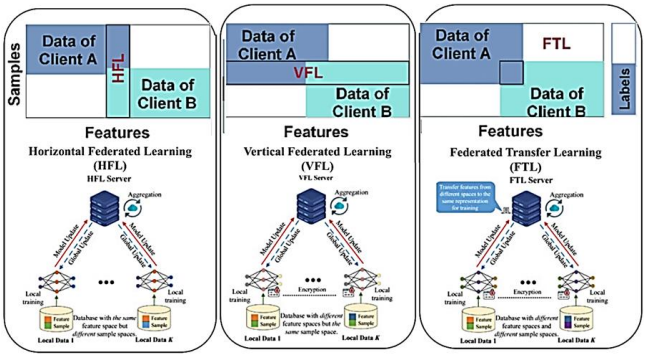


Figure 1. Three distinct types of FL approaches

The categorization of FL, based on data distribution among clients, is crucial to understanding its framework. As proposed in (Yang et al., 2019c) FL can be classified into Horizontal FL, Vertical FL, and Federated Transfer Learning, depending on the distribution of features and sample spaces. These frameworks and their data characteristics are briefly described in subsequent sections, followed by an exploration of the various data distributions prevalent in FL settings.

Despite its advantages, FL encounters significant challenges, particularly in handling non-independent and non-identically distributed (non-IID) data across clients. Real-world datasets often exhibit heterogeneity due to variations in user behaviors, local environments, or institutional practices. This heterogeneity can adversely impact convergence rates, degrade model performance, and exacerbate divergences between local and global models, thereby complicating collaborative learning. While the Federated Averaging (FedAvg) algorithm, introduced by (McMahan et al., 2017a), laid the groundwork for FL, its efficacy diminishes under non-IID conditions, necessitating the development of alternative strategies.

To address these challenges, researchers have proposed diverse methodologies, including personalized FL, communication-efficient techniques, and mechanisms to enhance robustness against adversarial threats (Mills et al., 2019), (Xu et al., 2020a), (Lim et al., 2020), (Kairouz et al., 2019), (Zhu et al., 2020). However, a comprehensive review addressing the impact of non-IID data on FL remains absent in the literature. This paper seeks to bridge this gap by offering an in-depth examination of non-IID data in FL, its effects on model training and aggregation, and advancements in mitigating associated challenges. Additionally, unresolved issues and future research directions are outlined, providing a holistic perspective on FL in non-IID settings.

(Kulkarni et al., 2020) briefly explored personalization methods to tackle non-IID data in FL. However, a systematic survey encompassing the definition, classification, and implications of non-IID data remains unexplored. This paper systematically examines: (1) the classification and characteristics of non-IID data; (2) its impact on model aggregation and performance in FL; (3) existing solutions and techniques to mitigate its challenges; and (4) limitations such as data bias and privacy concerns in current approaches. Finally, open problems and future directions for FL in non-IID contexts are discussed.

1.1. Horizontal Federated Learning

HFL has many potential applications in scenarios where data privacy and security are important, such as healthcare, finance, and social networks. However, HFL also faces some challenges, especially in terms of communication efficiency (Aledhari et al., 2020). Since HFL requires frequent exchange of model parameters between clients and a central server, it consumes a lot of bandwidth and energy resources. To address this issue, various methods have been proposed to reduce the communication overhead in HFL, such as:

- Subsampling of client updates: Instead of sending updates from all clients to the server, only a subset of clients is selected randomly or based on some criteria to participate in each round of communication (Arivazhagan et al., 2019), (Bonawitz et al., 2019)
- Model quantization: Instead of sending full-precision model parameters, only low-precision or compressed versions of them are transmitted, which can reduce the communication size and improve the robustness to noise (Bonawitz et al., 2017), (Briggs et al., 2020a).
- Communication frequency reduction: Instead of communicating every layer of a deep neural network model, only some layers are communicated selectively or periodically, which can exploit the heterogeneity and redundancy of different layers (Campos et al., 2017). Alternatively, some layers can be trained locally by each client using their own features and knowledge transferred from other layers.
- Multi-objective optimization: Instead of optimizing only one objective function (e.g., model accuracy), multiple objectives can be considered simultaneously (e.g., communication cost), and a trade-off solution can be found using evolutionary algorithms or other methods (Chang et al., 2019).

These methods can improve the communication-efficiency of HFL and make it more scalable and practical for real-world applications.

One challenge of HFL is that it requires weighted model averaging to update the global model, which has limited theoretical evidence to support its effectiveness. This may lead to suboptimal or even divergent global models, especially when parametric models are used in HFL. A possible solution is to use non-parametric models, such as kernel methods or neural networks, which can better capture the complex and nonlinear relationships among the features and the labels.

The privacy of the data used by each client in federated learning (FL) is not guaranteed by simply preventing direct access from third parties (Shokri and Shmatikov, 2015). In fact, it is possible to infer private image data from the gradient information of both shallow and deep neural networks that are uploaded by the clients (Wang et al., 2019b), (Zhao et al., 2020a). To address this issue, some privacy protection techniques such as homomorphic encryption (HE) and differential privacy (DP) have been proposed (Lim et al., 2020). (Phong et al., 2018) used network-based HE to encrypt the local model parameters before uploading them. (Hao et al., 2019) proposed a faster symmetric HE scheme to reduce the encryption time, while (Zhang et al., 2020a) used a quantization technique to encode the model parameters as a single vector and encrypt it as a whole, reducing the encryption time in horizontal FL. Recently, (Zhu et al., 2020) designed a distributed HE scheme suitable for horizontal FL, where the server and clients jointly generate key pairs. They also used ternary gradient quantization and an approximate aggregation method to speed up the encryption process, especially for deep neural networks.

1.2. Vertical Federated Learning

Vertical federated learning (FL) is a type of FL that deals with the scenario where different clients have data with the same labels but different features. This is also known as feature-based FL and is also called heterogeneous FL (Yu et al., 2020b), (Cheng et al., 2019).

Vertical federated learning (FL) with logistic regression (Montgomery et al., 2021) is a way of learning from data that is distributed across different parties without exposing their private information. It is useful for binary classification problems, where the goal is to predict one of two possible outcomes. In vertical FL, each party has some of the features for the same set of samples, but not all. They can work together to train a logistic regression model that uses all the features by exchanging intermediate results. Unlike horizontal FL, vertical FL does not need a central server or a global model. Each party has its own local model that fits its own data features, and they do not share these models with others. The guest party calculates the predictions and sends them to the host parties, who then update their local models with the partial gradients. The guest party also updates its own model with the gradients.

One of the key issues in vertical FL is privacy. Several methods have been proposed to address this challenge. For example, (Hardy et al., 2017) developed a scheme to separate identifiers securely for vertical logistic regression, and (Nock et al., 2018) studied the impact of ID segregation on the performance. (Yang et al., 2019d) proposed a simplified vertical FL framework that does not need a third-party coordinator. In contrast to these works that use parametric models for vertical FL, (Chen et al., 2024) used the xgboost decision tree model to design a secure system. (Wu et al., 2020) also used a decision tree model and introduced a scheme called Pivot, which is privacy-preserving and does not rely on any trusted third party.

Vertical FL has attracted much attention for its ability to protect data privacy while enabling collaborative learning among multiple parties. However, privacy protection is not the only challenge in vertical FL. Learning performance is also an important aspect that needs to be improved. Several methods have been proposed to enhance the efficiency and effectiveness of vertical FL. For example, (Yang et al., 2019a) applied the quasi-Newton method to speed up the convergence of vertical federated logistic regression. (Liu et al., 2019b) developed a FedBCD algorithm that reduces the communication rounds by allowing each party to perform multiple local updates before sharing. (Feng and Yu, 2020) introduced an MMVFL framework that can handle multi-layer problems with multiple parties in vertical FL. (Chen et al., 2020) utilized a perturbed local embedding technique to achieve both privacy preservation and communication-efficiency in asynchronous vertical FL.

A systematic review was conducted in accordance with the PRISMA guidelines. Relevant studies were identified through a comprehensive search of major databases, including IEEE Xplore, arXiv, and Google Scholar, using keywords such as "Federated Learning," "non-IID Data," and "Distributed Learning Challenges." The review included research articles, preprints, and review papers

published between 2016 and 2024. Articles were screened to identify those addressing the challenge of categorical non-IID data and the corresponding solutions proposed in the literature. A total of 85 studies met the inclusion criteria and were incorporated into the final analysis. The challenges identified in the literature and categorized in this study are presented in Figure 2.

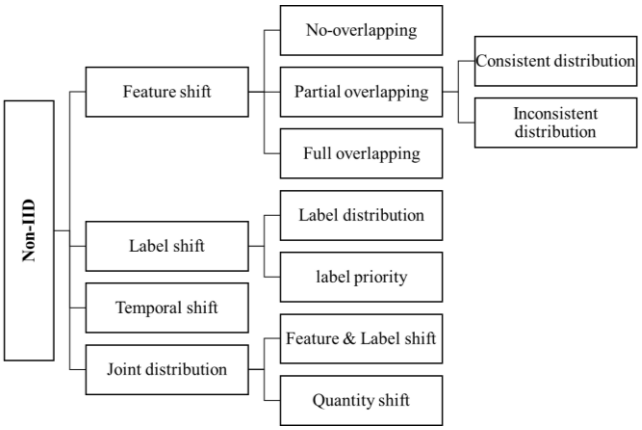


Figure 2. Categories of non-IID Data challenge in FL

2. Categories of non-IID Data

non-IID data is a common challenge in federated learning, where the data samples on different clients are not independent and identically distributed. This means that the local data distributions may vary significantly from each other and from the global data distribution. non-IID data can affect the performance and convergence of federated learning algorithms, especially for parametric models that rely on gradient-based optimization. In this section, we will introduce some types of non-IID data and their impact on horizontal federated learning (Lu et al., 2024).

Horizontal federated learning is a scenario where clients share the same feature space but have different labels. For example, different clients may have images of different classes or categories. In this case, we can classify non-IID data into three categories based on the perspective of feature x, label y and joint distribution p(x,y).

- non-IID in feature space (feature distribution skew or covariate shift): This type of non-IID data occurs when the feature distributions on different clients are different, but the label distributions are similar. For example, clients may have images with different resolutions, brightness or orientations, but the same classes. Feature non-IID data can cause gradient mismatch and slow down the convergence of federated learning.
- non-IID in label space (label distribution skew or prior probability shift): This type of non-IID data occurs when the label distributions on different clients are different, but the feature distributions are similar. For example, clients may have images of different classes, but with similar characteristics. Label non-IID data can cause model divergence and reduce the accuracy of federated learning.
- non-IID in joint distribution (p(x,y)): This type of non-IID data occurs when both the feature and label distributions on different clients are different. For example, clients may have images of different classes and different characteristics. Joint non-IID data can cause both gradient mismatch and model divergence, and pose a great challenge for federated learning.

The analysis revealed 4 primary categories of non-IID challenges:

2.1. Feature shift (Feature skew)

Feature shift is a term that describes the situation when the data feature of different clients has different distributions. For example, one client may have data that is mostly numerical, while another client may have data that is mostly categorical. Or, one client may have data that covers a wide range of values, while another client may have data that is concentrated around a narrow range. Feature shift can pose challenges for federated learning, which aims to train a global model using data from multiple clients. If the feature distributions are too different, the global model may not perform well on some clients or may not generalize well to new data.

Feature shift non-IID is a common problem in FL, such as different scanners/sensors in medical imaging or different scenery distribution in autonomous driving where local clients store examples with different distributions compared to other clients. To alleviate the feature shift before averaging models, local batch normalization can be used. The resulting scheme, called FedBN (Li et al., 2021), outperforms both classical FedAvg and the state-of-the-art for non-IID data (FedProx) (Li et al., 2020).

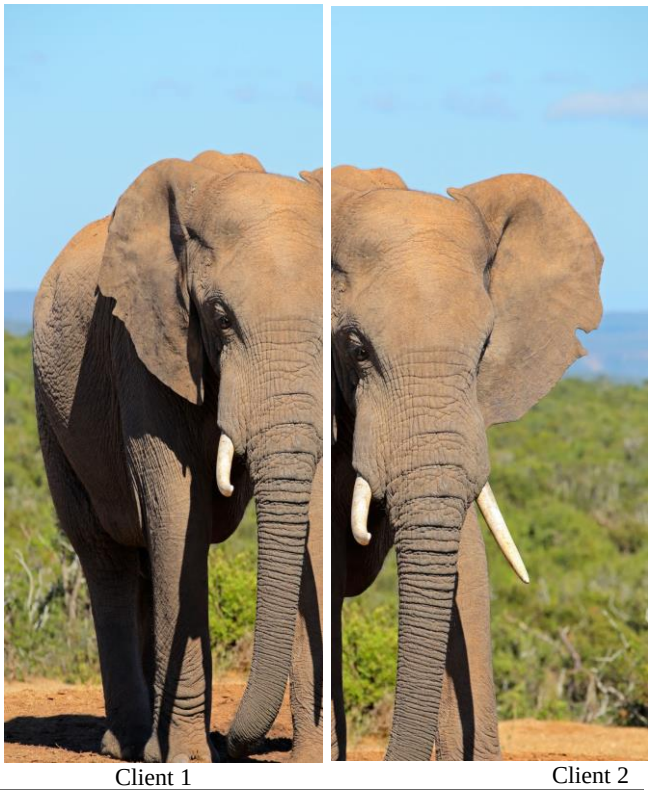
Non-overlapping feature shift

VFL is suitable for scenarios where data is vertically partitioned, meaning that different parties have different features for the same set of users. For example, a bank may have financial information of its customers, while an e-commerce platform may have their shopping preferences.

VFL can leverage the complementary information from different parties to improve the model performance.

One of the challenges in VFL is how to align the data samples from different parties based on their identifiers. A non-overlapping feature shift means that data feature is unique among clients. In other words, no two clients have the same feature for any user. In this case, if x data samples on k different clients with the same identifiers have the same labels, it is known as vertical FL. This non-overlapping property can guarantee that the calculated total loss of the logistic regression (linear model) in the vertical FL is equal to the focused learning.

To illustrate this concept, we can use two examples of datasets that are suitable for VFL: for elephant image data as shown in Figure 3, an elephant image is divided into two non-overlapping pieces, and client 1 stores the left part and client 2 stores the right part. The main difference between these two types of datasets is that adjacent features of personal information may not be related, but adjacent pixels for image data are always highly dependent.



Client 1 Client 2

Figure 3. An example of non-overlapping feature shift for image datasets, the left half of the image data is stored on Client 1, while the right half of the image is on Client 2.

Partial overlapping feature shift

The second type of feature shift is partial overlapping feature shift, where some feature of the data can be partially shared among different clients. For example, multi-view images which are captured from different perspectives, and each of them contains a single-view (single-perspective) image (Su et al., 2015). One way to represent 3D shapes is by using multi-view images, which are collections of single-view images taken from different angles. For example, Figure 4 shows a chair captured from 12 different perspectives. These images are fed into CNN1 to extract view-specific features, which are then aggregated across views and passed to CNN2 to produce a compact shape descriptor.

Note that the distribution of any shared feature between different clients may be either consistent or inconsistent.

Consistent distribution: In this scenario, the i-th shared feature is drawn from the same distribution, which implies that common feature does not increase the discrepancy of non-IID data.

Inconsistent distribution: This term refers to the scenario where the common features among different clients are not evenly distributed, leading to non-IID data divergence. For instance, considered the possibility that some clients may sample data from an unsuitable domain in the input space.

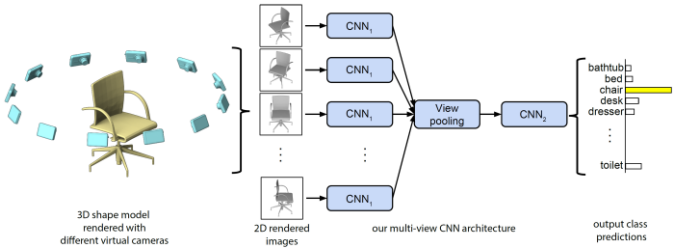


Figure 4. Two examples of feature distribution shift.

Full overlapping feature shift

HFL assumes that the clients have the same feature space, but different data samples. For example, different banks may have the same feature for their customers, but different customer records.

However, HFL faces some challenges when the data distributions are inconsistent among the clients. This may happen due to various factors, such as noise, outliers, or domain shifts. For instance, Figure 5 shows how different clients may have different levels of pulse noise in their data. Another example is the EMNIST dataset (Cohen et al., 2017), which contains handwritten digits from different people. Even for the same digit, the writing styles (e.g., width and slant) may vary across different clients.

These factors can cause divergence in the HFL process and degrade the model performance.

Figure 5. different clients may have different levels of pulse noise in their data

To address these challenges, some methods have been proposed to enhance HFL by leveraging the unique features of each client. These methods aim to exploit the diversity and complementarity of the client-specific features, while preserving the commonality of the shared features. (Mori et al., 2022), propose a novel HFL method using neural networks called continual horizontal federated learning (CHFL), which splits the network into two columns corresponding to common features and unique features, respectively. It jointly trains the first column by using common features through vanilla HFL and locally trains the second column by using unique features and leveraging the knowledge of the first one via lateral connections without interfering with the federated training of it.

2.2. Label shift (Label skew)

Label shift is a common challenge in federated learning (FL), where the label distribution varies across different clients. Label shift can be divided into two subtypes: label distribution shift and label priority shift.

Label distribution shift

Label distribution shift refers to the situation where the conditional feature distribution is shared among clients, but the marginal label distribution is different. This can happen when clients have different data collection settings or preferences that affect the label frequencies. For example, some clients may have more samples of certain classes than others.

One possible way to address label shift is to use logits calibration (Zhang et al., 2022), which adjusts the logits before softmax cross-entropy according to the probability of occurrence of each class. This can reduce the overfitting or underfitting of local models to minority or majority classes, and improve the accuracy and stability of the global model.

Another possible way to address label shift is to use label hashing (Dai et al., 2021), which compresses the original labels into binary codes using a hash function. This can reduce the model size and communication cost, as well as enhance the generalization ability and convergence rate of FL models.

label priority shift

Label priority shift refers to the situation where the marginal feature distribution is shared among clients, but the conditional label distribution is different. This can happen when clients have different labeling criteria or standards that affect the label assignments. For example, some clients may label images more strictly or loosely than others.

Different from label distribution shift, label priority shift takes into account the client data sample intersection issues often encountered in real-world applications, where the conditional distribution may vary across clients, even though it is the same.

If training data overlaps horizontally across clients, it is likely that different users will annotate different labels for the same data instance due to individual preference. This means that different users may have access to the same data instance and assign different labels to it based on their personal preferences. For example, YouTube users can express their opinions on videos by clicking the "Like" or "Dislike" buttons. However, these ratings are subjective and may vary from user to user. As shown in Figure 6, one user may like a video while another user may dislike it for the same video. This can create inconsistency and confusion for the algorithms that rely on the training data.

Crowdsourcing data (Garcia-Molina et al., 2016) is a more complex situation where data labels can be noisy and pose serious challenges for information gathering in many central machine learning tasks, let alone FL tasks. For example, most local devices only contain unlabeled data and require multiple users or volunteers to label them. Therefore, it is not uncommon for some tags to be inaccurate, noisy, or even lost.

Both types of label shift can degrade the performance of FL models, especially when the data is highly imbalanced or shifted. Therefore, it is important to design robust and adaptive FL algorithms that can handle label shift effectively.

Figure 6. one user may like YouTube video while another user may dislike.

2.3. Temporal shift

Temporal shift is a type of non-IID data distribution that occurs in federated learning applications involving temporal data, such as spatio-temporal data and time-series data. Temporal data are often collected by different devices at different time points or intervals, resulting in a shift in the temporal index of the data observations across clients. Temporal shift can affect the performance and convergence of FL models, especially for those that rely on the temporal correlation of the data, such as graph neural networks (GNNs).

One example of temporal shift is shown in Figure 7, where two webcams at different locations capture images everything over time, and the images shift in different clients due to the time difference. Although there is a temporal shift in the data collected by each Client, these data have a large number of intersections over the entire time period.

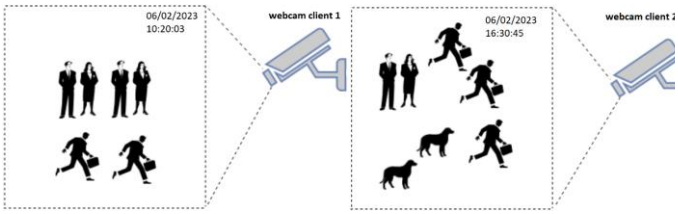


Figure 7. two webcams at different locations capture images over time.

To address the challenge of temporal shift in FL, some possible solutions are: 1) aligning the temporal index of the data across clients before federated training; 2) applying temporal-spatial transformation to the data to capture both the spatial and temporal features; 3) disentangling the domain-specific feature from the causal factors of the data and learning a shared representation for federated learning.

2.4. Joint distribution shift (Feature, Label)

There are other scenarios that do not fall into any of the non-IID categories discussed above.

Feature & Label shift

In the Feature & Label shift scenario, different clients store data with different labels and different feature, which integrates horizontal and vertical FL feature.

One of the challenges in federated learning is to handle heterogeneous data types among different clients. Data types refer to the characteristics and formats of the data, such as video, speech, text, image, etc. Different data types may require different models or algorithms to process and learn from. Therefore, a common definition of data types is needed to enable federated learning across diverse clients. Figure 8 illustrates an example of how data types can vary between clients. In this example, Camera has only video data and PC has text and speech data. These two data types are very different in terms of their features, dimensions, and representations. It is not easy to integrate a global model that can work well for both data types, as the local models may have different architectures or parameters depending on the data types.

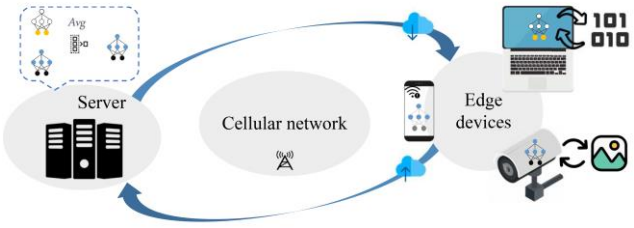


Figure 8. An example of different features, different labels non-IID . Camera and PC may hold different types of data.

Quantity shift

Quantity shift is a phenomenon that occurs when the distribution of data among different clients is uneven. For example, some clients may have more data than others, or some clients may have data from a specific domain or category. This can happen in any of the scenarios we have discussed so far, such as horizontal, vertical, or hybrid federated learning.

3. non-IID data management

non-IID datain federated learning, especially in horizontal FL where the data distribution on each client device may differ significantly from the global distribution is very important challenge. This may lead to poor

performance of the global model trained by aggregating local models. To address this issue, researchers have proposed various approaches that can be categorized into three types: data-based, algorithm-based, and system-based (Ma et al., 2022).

Data-based approaches aim to improve the data quality or diversity on each client device by applying data augmentation, data synthesis, or data sharing techniques. For example, some methods use generative models to create synthetic data that can enhance the local data representation. Other methods use a small fraction of globally shared data to reduce the data heterogeneity among clients.

Algorithm-based approaches aim to modify the federated learning algorithm to make it more robust or adaptive to non-IID data. For example, some methods use personalized learning techniques to fine-tune the global model on each client device according to the local data characteristics. Other methods use adaptive aggregation techniques to assign different weights or importance to each local model based on the data quality or similarity.

System-based approaches aim to optimize the system design or architecture to mitigate the impact of non-IID data. For example, some methods use clustering techniques to group clients with similar data distributions into sub-federations and train separate models for each sub-federation. Other methods use hierarchical federated learning techniques to organize clients into different levels of hierarchy and aggregate local models in a bottom-up manner.

The advantages and disadvantages of these methods will be discussed in detail. In this section, if not specified otherwise, FL stands for horizontal FL, the FL algorithm is the base FedAvg described in Algorithm 1, and the models are neural networks.

The solutions to the challenges discussed in the previous section, as identified in the literature, are illustrated in Figure 9.

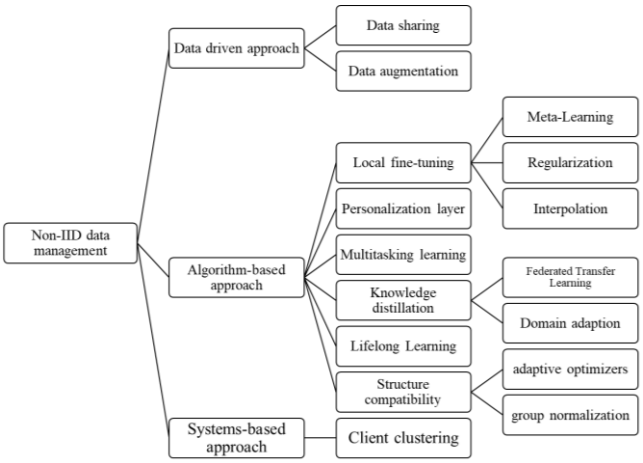


Figure 9. non-IID datamanagement approaches

3.1. *Data driven approach*

non-IID data poses a significant challenge for federated learning systems, as it affects the convergence and accuracy of the models trained on heterogeneous data sources. To cope with this problem, data-driven approaches aim to modify the data distributions by either sharing or augmenting the data among the local devices. Data sharing methods transfer a subset of data from one device to another to increase the overlap and diversity of the data. Data augmentation methods generate synthetic data based on existing data to enrich the data quality and quantity. Both methods have been shown to improve the performance of federated learning on non-IID data in various scenarios.

Data sharing

Data sharing is a simple yet effective technique to deal with non-IID data in HFL. As shown in Figure 10 in data sharing, a global dataset G with a uniform distribution is stored on the server and used to train the global model. Moreover, each client downloads a random fraction of G and updates the model with both its local data and the shared global data.

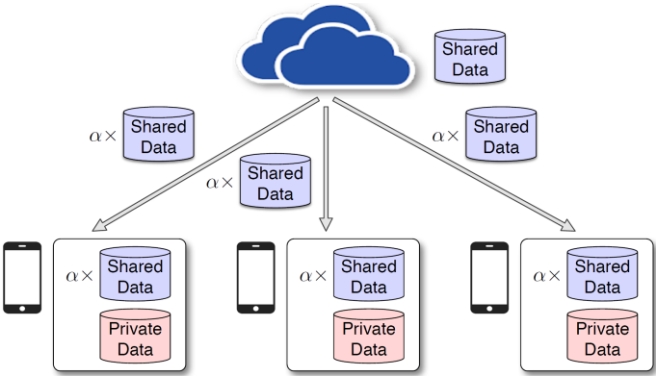


Figure 10. data sharing, a global dataset G with a uniform distribution and each client downloads a random

fraction of G and updates the model with both its local data and the shared global data

Similar approaches have been proposed in (Tuor et al., 2020) to reduce the negative impact of non-IID data by sharing some local data with the server. However, this data sharing method has some obvious drawbacks.

First, it is hard to obtain a uniformly distributed global dataset, because the server does not know the data distribution among the connected clients.

Second, downloading parts of the global dataset for each client violates the privacy-preserving requirement that is the main motivation of federated learning.

Data augmentation

Data augmentation is a technique to increase the diversity and quality of training data by applying some transformations or knowledge transfer, which can also help to reduce the data imbalance problem in federated learning. However, non-IID challenge can lead to biased and unfair global models with low accuracy and high variance across clients.

To address this issue, several data augmentation methods have been proposed for FL, such as vanilla method (Luca et al., 2022), mixup method (Shin et al., 2020), and generative adversarial network (GAN) based method (Rasouli et al., 2020).

The vanilla method is the simplest one, where each client creates redundant copies of its local data samples with some random noise or perturbation. These data augmentation methods can improve the performance and fairness of FL by mitigating the data heterogeneity and increasing the data representation among clients.

The mixup method is a more advanced one, where each client mixes its local data samples with other samples from different classes or domains using a convex combination. The main idea is that each client encrypts its data samples using the XOR operator and sends them to the server for decryption. The server then combines the original and the decrypted data samples to create a new balanced data set.

The GAN-based method is the most sophisticated one, where each client uses a GAN model to generate synthetic data samples that are similar to its local data distribution. Unlike the previous two methods, federated GAN data augmentation aims to train a good generator in the presence of non-IID data. The synthetic data can help each client to fill the gaps in its local data and create a more balanced dataset. The federated GAN data augmentation method can be divided into two categories: centralized and decentralized. In the centralized approach, the server collects some seed samples from each client and trains a global GAN model based on them. Then, the server sends the trained generator to each client, and each client uses it to augment its local data. However, this approach may violate the data privacy of the clients and reveal their label distribution information to the server. In the decentralized approach, each client trains a local discriminator and uploads it to the server. The server aggregates the local discriminators to form a global discriminator, and updates the generator using its loss gradients. Then, the server sends the updated generator back to each client, and each client uses it to augment its local data. This approach can preserve both data privacy and label distribution information of the clients, but it may require more communication rounds and computation resources.

3.2. Algorithm-based approach

One possible way to address the challenges of federated learning is to adopt an algorithm-based approach that adapts the learning process to the characteristics of the local data and the global model. In this approach, each client can customize its local model according to its own data distribution, preferences, and constraints, while still contributing to the global model. This can be achieved by using different optimization techniques, such as gradient-based methods, meta-learning, or reinforcement learning, that can balance the trade-off between local and global objectives. Algorithm-based approaches can also leverage communication-efficient strategies, such as compression, sparsification, or quantization, to reduce the communication overhead and latency of federated learning. Algorithm-based approaches can potentially improve the performance, robustness, and scalability of federated learning systems (Yurochkin et al., 2019).

Personalization is a way of adapting a model to local tasks. It can help with some issues that arise when using a general model. There are different kinds of personalization methods, such as:

- **Local fine-tuning**: This means adjusting the model parameters with regularization, interpolation, or meta-learning.
- **Personalization layer**: This means adding a layer to the model that is specific to each task.
- **Multi-task learning**: This means training the model on multiple tasks at the same time.

- **Knowledge distillation**: This means transferring knowledge from a larger model to a smaller one (Deng et al., 2020).

Local fine-tuning

Local fine-tuning is a widely used personalization technique that adapts the local models to the client-specific data after receiving a global model from the server, and FedAvg is a simple and effective form of local fine-tuning. The goals of local fine-tuning are to find a good initial global model that can be easily adapted to different clients, and to combine the local and global information for better performance.

Meta-Learning

One common approach for finding a good initial global model is to use meta-learning methods, such as MAML (Finn et al., 2017), that can learn a model that is suitable for fast adaptation. A representative method that uses this approach is Per-FedAvg (Fallah et al., 2020), which leverages MAML to find an initial global model that can achieve good performance with low computation costs on the clients In Per-FedAvg. (Jiang et al., 2019), who combined FedAvg with Reptile (Nichol et al., 2018), a meta-learning algorithm that learns a good initialization for fast adaptation, and (Chen et al., 2018), who proposed a federated meta-learning framework that shares meta-learning across clients. They argue that their method is different from multi-task learning and collaborative meta-learning, which may pose privacy risks for users.

Regularization

The idea of regularization is to reduce the discrepancy between the local and global models, and FedAvg can be seen as a special case of regularization-based personalization. (Hanzely and Richtárik, 2020) proposed a new formulation of the objective function that includes a regularization term to balance the trade-off between local and global models. (Dinh et al., 2020). applied (Moreau, 1963) Envelopes to their pFedMe algorithm to deal with the statistical heterogeneity among clients. They added an l_2 norm term to the client's objective function to penalize the deviation from the global model. (Huang et al., 2021) customized FL with additional expressions and a focused attention messaging (FedAMP) strategy to reduce the effect of non-IID data, and they guarantee the convergence of the proposed FedAMP and obtain satisfactory results on several widely used datasets.

Interpolation

Data interpolation combines local and global data for training, while model interpolation combines local and global models as a personalized model. (Mansour et al., 2020) conducted a systematic empirical study on three personalization strategies, client clustering, data interpolation, and model interpolation, as well as their theoretical guarantees. However, this method requires the investigation of challenging tasks.

Personalization layer

FedPer is a novel federated learning method that leverages shallow neural networks layer for feature extraction and deep networks layer for personalization. This idea is based on (Arivazhagan et al., 2019) who showed that shallow layers are sufficient for high-level feature representation in classification tasks. Unlike FedAvg, which updates all layers equally, FedPer updates the personalization layers independently. FedPer can achieve much higher test accuracy than FedAvg on strongly non-IID data, according to experimental results. Interestingly, FedPer performs better on non-IID datathan on IID data.

Liang et al. proposed LG-FEDAVG (Liang et al., 2020), a new federated learning method that learns both local and global representations for each device. The local representations are device-specific features learned by shallow neural networks, while the global representation is a shared deep neural network that performs classification.

Multitasking learning

Another way to address the personalization challenge is to formulate it as a multi-task learning problem. For instance, MOCHA (Smith et al., 2017), a representation-based framework for federated multi-task learning (FMTL), initially tackles the issues of communication overhead, stragglers, and fault tolerance for FL. However, MOCHA generates separate but correlated models for each client, which makes it unsuitable for non-convex optimization problems. (Corinzia et al., 2019) proposed a virtual FMTL framework using a Bayesian network and inferring approximate updates that can handle non-convex models. Their method has achieved promising results on several non-IID datasets, but it has difficulty when converging with a large number of clients due to sequential fine-tuning. To mitigate FL performance degradation due to heterogeneous data distribution, (Sattler et al., 2020) proposed a non-convex FMTL framework, called cluster federated learning (CFL), for clustering local data. CFL provides a computationally efficient metric of client population distribution based on cosine similarity and has

obtained significant results on non-IID data. However, CFL may introduce new challenges for data security due to its dependence on data similarity.

Knowledge distillation

Knowledge distillation is a technique that allows smaller models to learn from larger models in FL. It can help overcome the challenges of non-IID data distribution and improve the performance of personalized models. There are two main approaches to knowledge distillation in federated learning: federated transfer learning and domain adaption (Mora et al. 2022).

Federated Transfer Learning

Federated transfer learning aims to transfer the knowledge from a global model or other clients to a specific client, using methods such as homomorphic encryption, secret sharing, or hyperparameter tuning. For example, (Liu et al., 2020a) proposed a federated transfer learning framework that ensures secure and efficient modeling under decentralized and fragmented data. (Lin et al., 2020) used a group distillation strategy to combine multiple models into a single distilled model that reduces the risk of leakage and the computational cost.

First strategy is to use unlabeled data or synthetic samples to enhance the knowledge transfer among the parties. For example, (Zhang et al., 2018b) describe a method that uses unlabeled data or generated synthetic samples to help extract knowledge from all participating clients. This method can be applied to both homogeneous and heterogeneous settings, although unlabeled data used may lead to increased computing budget.

Second strategy is to use feature extraction or representation learning to reduce the dimensionality and complexity of the data. For example, (Chang et al., 2019). proposed Kronos a collaborative reinforcement learning method by uploading trained features instead of local models to implement local personalization. This method can reduce the communication overhead and improve the generalization ability of the model.

Third strategy is to use peer-to-peer communication or decentralized coordination to avoid relying on a central server or authority. For example, (Li et al., 2020a). proposed a framework called decentralized federated learning through mutual knowledge transfer (Def-KT), in which local clients exchange messages directly on a peer-to-peer basis without the involvement of a cloud server. The authors state that the performance degradation of FedAvg in heterogeneous data may be caused by only moving model parameters, and the key point of Def-KT is to take advantage of mutual knowledge transfer (MKT) to reduce the effect of label shift. Specifically, during each communication round, a subset of selected clients first trains their models locally and then transmit the updated models to a second subset of clients. Then the clients in the second subset can calculate two soft predictors (logits) based on their local models and get the trained models. These two computed logits are used as dummy labels to update both the local model and the received model at each client in the second subset.

Domain adaption

Domain adaption aims to align the domains of different clients or the server and the clients, using methods such as domain adaptation, domain generalization, or meta-learning. For example, (Hinton et al., 2015) proposed a domain adaption federated learning framework that leverages domain adaptation techniques to match the source and target domains of different clients. (Wang et al., 2019a) proposed a meta-learning based federated learning framework that learns a meta-model that can be quickly adapted to new devices for next word prediction.

One way to improve the efficiency of knowledge transfer between models is to use domain adaption, which aims to reduce the discrepancy between the data distributions of different clients. A novel method for domain adaption in federated learning (FL) systems is the federated adversarial domain adaption algorithm (FADA) proposed in (Peng et al., 2019). FADA leverages adversarial learning techniques to align the feature distributions of different clients in a common latent space. Another approach for domain adaption in FL is FedMD, introduced by (Li and Wang, 2019) in FedMD allows each client to train a customized model on its local data, while benefiting from the knowledge distilled from a public dataset shared by all clients. This public dataset does not pose any privacy risk and can be used to train an initial model for each client. For instance, a client can first train a model on a subset of CIFAR100 (public dataset) and then fine-tune it on CIFAR10 (private dataset).

Lifelong Learning

Lifelong learning is a crucial skill for machine learning practitioners, who need to train models on sequential tasks without access to previous data. The main challenge is to preserve the performance of the model across different tasks and avoid catastrophic forgetting. Therefore, the concept of lifelong learning can be applied to deal with non-IID dataissues.

Elastic Weight Consolidation (EWC) is a powerful technique to mitigate catastrophic forgetting in lifelong learning, where the most relevant parameters for a given task A are determined. When the model is trained on a new task B, the learner is penalized for modifying these parameters. Drawing an analogy between federated learning and lifelong learning, (Shoham et al., 2019) propose an EWC-based federated curvature (FedCurv) algorithm as a solution to non-IID problems in FL. In each round, participants send updated models along with the trace of the Fisher information matrix, which represents the most informative parameters for the current task. A penalty term is added to the loss function for each participant to encourage convergence towards a global joint optimum. Moreover, they assume that the communication cost can be further reduced by a quantized version of the uploaded parameters, without empirical validation. (Liu et al., 2019a) combined lifelong learning and reinforcement learning to form a federated reinforcement learning lifelong (LFRL) architecture. With LFRL, they enable robots to integrate and transfer experience, so that the robot can quickly adapt to a new environment.

Structure compatibility

As we have seen, FL faces many difficulties when dealing with complex models like DNNs, especially when the data is non-IID. In this section, we will present and analyze two effective methods to improve the convergence rate of DNNs in FL.

adaptive optimizers

The first method is to adopt adaptive optimizers such as Adagrad (Duchi et al., 2011), Adam, and others, instead of the standard SGD optimizer. However, these adaptive optimizers usually need to store and update the historical gradient information to adjust the learning rate, which may increase the communication overhead in FL. This is because the local models are trained only on the edge devices and the historical gradients (which have the same size as the model parameters) also need to be sent to the server for aggregation. To overcome this issue, (Reddi et al., 2020) propose a federated adaptive scheme that simplifies the computation and communication. The main idea is straightforward: the historical gradients are computed based on the average global gradients on the server, and each client performs standard SGD to update the local model. This can save local computation resources and reduce communication costs, since the server is assumed to have much more computing power than the edge devices.

group normalization

Another useful method is to use group normalization (GN) layers (Wu and He, 2018) to replace the batch normalization (BN) layers (Ioffe and Szegedy, 2015) in deep neural networks (DNNs). BN layers normalize the features by the mean and variance computed within a mini-batch of data, which can speed up training and reduce the sensitivity to network initialization. However, BN layers require a sufficiently large batch size to work well, which limits the memory efficiency and the applicability to tasks with small batches, such as object detection, segmentation, and video analysis. GN layers overcome this limitation by dividing the channels of each data sample into groups and normalizing within each group independently. GN layers are independent of batch sizes and can achieve stable and consistent performance across a wide range of batch sizes. Experimental results have shown that GN layers can outperform BN layers for various computer vision tasks, such as image classification, object detection, segmentation, and video classification.

3.3. Systems-based approach

System level optimization is a crucial aspect of federated learning, which is a distributed learning paradigm that enables collaborative learning from decentralized data while preserving privacy. FL faces two main challenges: (1) systems heterogeneity, which refers to the variability of the devices' characteristics and communication capabilities in the network, and (2) statistical heterogeneity, which refers to the non-identical distribution of the data across the devices. To address these challenges, various FL algorithms have been proposed and implemented in different frameworks and platforms. In this section, we briefly review some of the existing works on system level optimization for FL.

However, measuring the data similarity between clients without violating their privacy is a difficult problem. Two main methods have been proposed: loss value similarity and model weights similarity .The first similarity evaluation approach based on loss values of cluster models is proposed in (Ghosh et al., 2019).

The main idea is that the server creates multiple global models and distributes them to the clients for computing local losses. Then, each client selects and updates the cluster model with the lowest loss and sends it back to the server for aggregation. This approach is implemented by the Iterative Federated Clustering Algorithm (IFCA) (Ghosh et al., 2020). IFCA uses a one-hot vector to indicate the

cluster assignment of each local model, and has two equivalent options for updating global models. A drawback of IFCA is that it requires clients to compute losses for all cluster models in every round, which increases the communication cost by a factor of k compared to FedAvg. A solution to this problem is suggested by (Kopparapu and Lin, 2020b), who introduce a fork algorithm that performs clustering only in certain rounds, thus reducing the clustering frequency.

The second similarity evaluation approach based on the local model weights. First, the global model is trained and warm up with FedAvg and then sent to each device. The device updates the model locally (gradients) and returns it to the server. The server calculates the similarity scores, such as cosine similarity, based on the received model weights and clusters the clients accordingly.

(Sattler et al., 2020) proposed a novel tree-based structure as shown in Figure 11. In this structure, a root node contains the Federated Learning model, which is trained on all clients {1…, m} and reaches a stationary point θ^*. The next layer splits the clients into two groups based on their cosine similarities and trains two sub-models that reach stationary points θ_0^* and θ_1^*. The splitting process continues until no sub-model meets the criteria. To assign new clients to a leaf model, the server stores the weight-updates Δ_e of each client before splitting at each edge of the tree. Then, the new client can follow the path with the highest similarity to find a suitable leaf model.

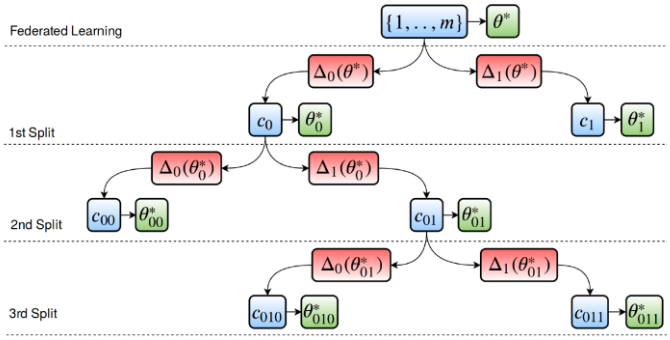


Figure 11. An exemplary parameter tree created by cluster FL (Kopparapu and Lin, 2020b).

In summary, system level optimization is an important research topic for FL, as it aims to improve the efficiency, scalability, and reliability of FL in heterogeneous data. There are many existing works on this topic, as well as many open challenges and opportunities for future research.

4. Future Directions

non-IID datadistributions in FL are a common and challenging scenario that affect the convergence and performance of federated learning models. In this section, we briefly summarize some of the main issues and possible solutions for dealing with non-IID datain FL.

Privacy protection is a fundamental and critical goal of FL, but many methods designed to deal with non-IID datasuch as data sharing, knowledge distillation inevitably increase the risk of privacy exposure. It is still unclear to what extent these methods compromise data privacy, and there are no quantitative measures to detect the extent of privacy leakage. One possible direction is to use “federated tabular data augmentation”, which can synthesize tabular data for data augmentation using some simple statistics (e.g., distributions of each column and global covariance) without sharing raw data or generative models.

FL contains a large number of hyperparameters, for example, the total number of clients, the number of local periods, and the client dropout probability, which vary greatly from one algorithm to another, making it difficult to benchmark the actual non-IID performance of these algorithms. A possible solution is to use “adaptive hyperparameter tuning” methods, which can adjust the hyperparameters dynamically according to the feedback from the clients and the server.

Although two real image datasets have been introduced in (Luo et al., 2019), (Hsu et al., 2020), a global homogeneous and heterogeneous benchmark dataset does not yet exist for FL. The generation of synthetic non-IID databy arbitrary partitioning datasets cannot effectively evaluate the performance of the proposed method for handling non-IID data. A possible direction is to “collect and publish more real-world datasets” that reflect the characteristics of non-IID datain FL, such as user preferences, device heterogeneity, and network conditions.

Personalized federated learning is promising for IoT edge devices, although it has not yet received sufficient attention. System design, model deployment, communication cost reduction in unstable and constrained wireless networks, and task adaptation with limited computing budget are still an open question. A possible solution is to use “federated meta-learning”, which can leverage the meta-knowledge from previous tasks to quickly adapt to new tasks with few-shot learning.

VFL is especially useful for scenarios where the data is feature-partitioned, meaning that different parties have different subsets of features for the same samples.

However, VFL faces many challenges, such as privacy preservation, communication efficiency, and non-IID data. non-IID datacan cause performance degradation and convergence issues for VFL algorithms. Therefore, it is important to investigate the impact of non-IID data on VFL and design effective methods to mitigate it.

One common type of non-IID data in VFL is overlapping data features, which means that some parties may share some common features with others. This can lead to redundant information and inconsistent gradients during model aggregation. Only a few works have attempted to address this problem by applying feature selection or feature transformation techniques. However, these methods may not be optimal or scalable for high-dimensional or complex data. Therefore, more research is needed to develop efficient and robust methods for dealing with overlapping data features in VFL.

Another type of non-IID data in VFL is different features and labels, which means that some parties may have different label spaces or different label distributions from others. This can result in label mismatch and model divergence during model aggregation. This type of non-IID data is more challenging than overlapping data features, as it requires more coordination and communication among the parties. Moreover, this type of non-IID data may also involve privacy issues, as some parties may not want to reveal their label information to others. Therefore, more research is needed to design secure and effective methods for dealing with different features and labels in VFL.

A third type of non-IID data in VFL is crowd shift, which means that the target distribution of the model may change over time due to dynamic environments or user preferences. This can result in model drift and performance degradation for VFL algorithms. This type of non-IID data is more challenging than the previous two types, as it requires more adaptability and flexibility for the model. Moreover, this type of non-IID data may also involve communication issues, as some parties may not be able to update their models frequently or synchronously with others. Therefore, more research is needed to design adaptive and efficient methods for dealing with crowd shift in VFL.

In summary, non-IID data is a common and critical issue for VFL that deserves more attention and investigation. This review examines existing research on non-IID data challenges in vertical and horizontal federated learning (VFL and HFL) settings, highlighting key findings and proposing potential directions for future work.

5. Conclusion

This paper aims to offer a systematic understanding of non-IID datain federated learning systems and to review the existing techniques for managing non-IID data. We provide a detailed classification of non-IID data distributions with illustrative examples, some of which have not been discussed in the literature. We highlight that non-IID datadistributions mainly affect the learning performance of parametric models in HFL and complex models such as DNN sensitive to client data distribution. We show that some existing works on non-IID data management, such as local fine-tuning and data sharing, often achieve better convergence performance at the cost of increasing local computation and communication resources or even compromising data privacy. Other methods, such as personalization and client clustering, require modification of the vanilla FL framework, and it is no longer possible to generate a global model for all clients. Finally, we discuss the remaining challenges in the management of non-IID in FL and suggest some research directions to address open questions.

non-IID dataposes various challenges and opportunities for future research in FL. For example, preserving privacy is a fundamental goal of FL, but some methods that address non-IID data, such as data sharing and knowledge distillation, may compromise privacy. Moreover, FL involves many hyperparameters, which makes it difficult to evaluate the true non-IID performance of these methods. Additionally, there is a growing need for automated machine learning (AutoML), but few studies have considered the effect of non-IID distributions. To cope with non-IID data, uses deep reinforcement learning to dynamically adjust the influence of each client.

Authors' Contributions

All authors equally contributed to this study.

Declaration

Not applicable.

Transparency Statement

Acknowledgments

Declaration of Interest

The authors report no conflict of interest.

Funding

According to the authors, this article has no financial support.

Ethical Considerations

Not applicable.

References

Yang, Q., Liu, Y., Chen, T., Tong, Y., (2019) c. Federated machine learning: Concept and applications. ACM Transactions on Intelligent Systems and Technology (TIST) 10, 1–19. https://doi.org/10.48550/arXiv.1902.04885

McMahan, B., Moore, E., Ramage, D., Hampson, S., y Arcas, B.A., (2017) a. Communication-efficient learning of deep networks from decentralized data, in: Artificial Intelligence and Statistics, pp. 1273–1282. https://doi.org/10.48550/arXiv.1602.05629

Mills, J., Hu, J., Min, G., (2019). Communication-efficient federated learning for wireless edge intelligence in iot. IEEE Internet of Things Journal 7, 5986–5994. https://doi.org/10.1109/JIOT.2019.2956615

Xu, J., Du, W., Jin, Y., He, W., Cheng, R., (2020) a. Ternary compression for communication-efficient federated learning. IEEE Transactions on Neural Networks and Learning Systems. https://doi.org/10.1109/TNNLS.2020.3041185

Lim, W.Y.B., Luong, N.C., Hoang, D.T., Jiao, Y., Liang, Y.C., Yang, Q., Niyato, D., Miao, C., (2020). Federated learning in mobile edge networks: A comprehensive survey. IEEE Communications Surveys & Tutorials 22, 2031–2063. https://doi.org/10.48550/arXiv.1909.11875

Kairouz, P.,McMahan, H.B., Avent, B., Bellet, A., Bennis, M., Bhagoji, A.N., Bonawitz, K., Charles, Z., Cormode, G., Cummings, R., et al., (2019). Advances and open problems in federated learning. arXiv:1912.04977. https://doi.org/10.48550/arXiv.1912.04977

Zhu, H.,Wang, R., Jin, Y., Liang, K., Ning, J., (2020). Distributed additive encryption and quantization for privacy preserving federated deep learning. arXiv:2011.12623. https://doi.org/10.48550/arXiv.2011.12623

Kulkarni, V., Kulkarni, M., Pant, A., (2020). Survey of personalization techniques for federated learning, in: 2020 Fourth World Conference on Smart Trends in Systems, Security and Sustainability (WorldS4), IEEE. pp. 794–797. doi: 10.1109/WorldS450073.2020.9210355

Aledhari, M., Razzak, R., Parizi, R.M., Saeed, F., (2020). Federated learning: A survey on enabling technologies, protocols, and applications. IEEE Access 8, 140699–140725. doi: 10.1109/ACCESS.2020.3013541

Arivazhagan, M.G., Aggarwal, V., Singh, A.K., Choudhary, S., (2019). Federated learning with personalization layers. arXiv:1912.00818. https://doi.org/10.48550/arXiv.1912.00818

Bonawitz, K., Eichner, H., Grieskamp, W., Huba, D., Ingerman, A., Ivanov, V., Kiddon, C., Konečn`y, J., Mazzocchi, S., McMahan, H.B., et al., (2019). Towards federated learning at scale: System design. arXiv:1902.01046. https://doi.org/10.48550/arXiv.1902.01046

Bonawitz, K., Ivanov, V., Kreuter, B., Marcedone, A., McMahan, H.B., Patel, S., Ramage, D., Segal, A., Seth, K., (2017). Practical secure aggregation for privacy-preserving machine learning, in: proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, pp. 1175–1191. https://doi.org/10.1145/3133956.3133982

Briggs, C., Fan, Z., Andras, P., (2020) a. Federated learning with hierarchical clustering of local updates to improve training on non-iid data, in: 2020 International Joint Conference on Neural Networks (IJCNN), IEEE. pp. 1–9. https://doi.org/10.48550/arXiv.2004.11791

Campos, V., Sastre, F., Yagües, M., Bellver, M., Giró-i Nieto, X., Torres, J.,(2017). Distributed training strategies for a computer vision deep learning algorithm on a distributed gpu cluster. Procedia Computer Science 108, 315–324. doi:10.1016/j.procs.2017.05.074

Chang, H., Shejwalkar, V., Shokri, R., Houmansadr, A., (2019). Cronus: Robust and heterogeneous collaborative learning with black-box knowledge transfer. arXiv:1912.11279. https://doi.org/10.48550/arXiv.1912.11279

Shokri, R., Shmatikov, V., (2015). Privacy-preserving deep learning, in: Proceedings of the 22nd ACM SIGSAC conference on computer and communications security, pp. 1310–1321. doi:10.1109/ALLERTON.2015.7447103

Wang, Z., Song, M., Zhang, Z., Song, Y.,Wang, Q., Qi, H., (2019)b. Beyond inferring class representatives: User-level privacy leakage from federated learning, in: IEEE INFOCOM 2019-IEEE Conference on Computer Communications, IEEE. pp. 2512–2520. https://doi.org/10.48550/arXiv.1812.00535

Zhao, B., Mopuri, K.R., Bilen, H., (2020) a. iDLG: Improved deep leakage from gradients. arXiv:2001.02610. https://doi.org/10.48550/arXiv.2001.02610

Phong, L.T., Aono, Y., Hayashi, T., Wang, L., Moriai, S., (2018). Privacy-preserving deep learning via additively homomorphic encryption. IEEE Transactions on Information Forensics and Security 13, 1333–1345. doi: 10.1109/TIFS.2017.2787987

Hao, M., Li, H., Xu, G., Liu, S., Yang, H., (2019). Towards efficient and privacy-preserving federated deep learning, in: ICC 2019-2019 IEEE International Conference on Communications (ICC), IEEE. pp. 1–6. doi:10.1109/ICC.2019.8761267

Zhang, C., Li, S., Xia, J., Wang, W., Yan, F., & Liu, Y. (2020). BatchCrypt: Efficient Homomorphic Encryption for Cross-Silo Federated Learning. *USENIX Annual Technical Conference*. USENIX Association. pp. 493–506. ISBN:978-1-939133-14-4

Yu, F., Zhang, W., Qin, Z., Xu, Z., Wang, D., Liu, C., Tian, Z., Chen, X., (2020)b. Heterogeneous federated learning. arXiv:2008.06767. https://doi.org/10.48550/arXiv.2008.06767

Cheng, K., Fan, T., Jin, Y., Liu, Y., Chen, T., Yang, Q., (2019). SecureBoost: A Lossless Federated Learning Framework. arXiv:1901.08755. https://doi.org/10.48550/arXiv.1901.08755

Montgomery, D.C., Peck, E.A., Vining, G.G., (2021). Introduction to linear regression analysis. John Wiley & Sons. ISBN: 978-1-119-57872-7

Hardy, S., Henecka, W., Ivey-Law, H., Nock, R., Patrini, G., Smith, G., Thorne, B., (2017). Private federated learning on vertically partitioned data via entity resolution and additively homomorphic encryption. arXiv:1711.10677. https://doi.org/10.48550/arXiv.1711.10677

Nock, R., Hardy, S., Henecka, W., Ivey-Law, H., Patrini, G., Smith, G., Thorne, B., (2018). Entity resolution and federated learning get a federated resolution. arXiv:1803.04035. https://doi.org/10.48550/arXiv.1803.04035

Yang, S., Ren, B., Zhou, X., Liu, L., (2019) d. Parallel distributed logistic regression for vertical federated learning without third-party coordinator. arXiv:1911.09824. https://doi.org/10.48550/arXiv.1911.09824

Tianqi Chen, Tong He, (20240. xgboost: eXtreme Gradient Boosting. Package Version: 1.7.8.1.https://cran.r-project.org/web/packages/xgboost/vignettes/xgboost.pdf

Wu, Y., Cai, S., Xiao, X., Chen, G., Ooi, B.C., (2020). Privacy preserving vertical federated learning for tree-

based models. arXiv:2008.06170. https://doi.org/10.14778/3407790.3407811
Yang, K., Fan, T., Chen, T., Shi, Y., Yang, Q., (2019) a. A quasi-newton method based vertical federated learning framework for logistic regression. arXiv:1912.00513. https://doi.org/10.48550/arXiv.1912.00513
Liu, Y., Kang, Y., Zhang, X., Li, L., Cheng, Y., Chen, T., Hong, M., Yang, Q., (2019) b. A communication efficient collaborative learning framework for distributed features. arXiv:1912.11187. https://doi.org/10.48550/arXiv.1912.11187
Feng, S., Yu, H., (2020). Multi-participant multi-class vertical federated learning. arXiv:2001.11154. https://doi.org/10.48550/arXiv.2001.11154
Chen, T., Jin, X., Sun, Y., Yin, W., (2020). VAFL: a method of vertical asynchronous federated learning. CoRR abs/2007.06081. URL: https://arxiv.org/abs/2007.06081, arXiv:2007.06081. https://doi.org/10.48550/arXiv.2007.06081
Z. Lu, H. Pan, Y. Dai, X. Si and Y. Zhang, "Federated Learning with Non-IID Data: A Survey," in IEEE Internet of Things Journal, vol. 11, no. 11, pp. 19188-19209, 1 June1, (2024), doi:10.1109/JIOT.2024.3376548.
Xiaoxiao Li, Meirui Jiang, Xiaofei Zhang, Michael Kamp, Qi Dou, (2021). FedBN: Federated Learning on Non-IID Features via Local Batch Normalization, https://doi.org/10.48550/arXiv.2102.07623
Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, Virginia Smith, (2020). Federated Optimization in Heterogeneous Networks, https://doi.org/10.48550/arXiv.1812.06127
Su, H.,Maji, S., Kalogerakis, E., Learned-Miller, E., (2015). Multi-view convolutional neural networks for 3d shape recognition, in: Proceedings of the IEEE international conference on computer vision, pp. 945–953. doi: 10.1109/ICCV.2015.114
Cohen, G., Afshar, S., Tapson, J., van Schaik, A., (2017). EMNIST: an extension of MNIST to handwritten letters. arXiv:1702.05373. https://doi.org/10.48550/arXiv.1702.05373
Junki Mori, Isamu Teranishi, Ryo Furukawa, (2022). Continual Horizontal Federated Learning for Heterogeneous Data. arXiv:2203.02108v2 https://doi.org/10.1109/IJCNN55064.2022.9892815
Jie Zhang, Zhiqi Li, Bo Li, Jianghe Xu, Shuang Wu, Shouhong Ding, Chao Wu, (2022). Federated Learning with Label Distribution Skew via Logits Calibration, https://doi.org/10.48550/arXiv.2209.00189
Zhenwei Dai, Chen Dun, Yuxin Tang, Anastasios Kyrillidis, Anshumali Shrivastava, (2021). Federated Multiple Label Hashing (FedMLH): Communication Efficient Federated Learning on Extreme Classification Tasks, https://doi.org/10.48550/arXiv.2110.12292
Garcia-Molina, H., Joglekar, M., Marcus, A., Parameswaran, A., Verroios, V., (2016). Challenges in data crowdsourcing. IEEE Transactions on Knowledge and Data Engineering 28(4):1-1, 901–911. doi:10.1109/TKDE.2016.2518669
Xiaodong Ma, Jia Zhu, Zhihao Lin, Shanxuan Chen, Yangjie Qin, (2022). A state-of-the-art survey on solving non-IID data in Federated Learning, https://doi.org/10.1016/j.future.2022.05.003
Tuor, T., Wang, S., Ko, B.J., Liu, C., Leung, K.K., (2020). Overcoming noisy and irrelevant data in federated learning. arXiv e-prints, arXiv–2001. https://doi.org/10.48550/arXiv.2001.08300
Artur Back de Luca, Guojun Zhang, Xi Chen, Yaoliang Yu, (2022). Mitigating Data Heterogeneity in Federated Learning with Data Augmentation, https://doi.org/10.48550/arXiv.2206.09979
Shin,M., Hwang, C., Kim, J., Park, J., Bennis, M., Kim, S.L., (2020). Xor mixup: Privacy-preserving data augmentation for one-shot federated learning. ArXiv abs/2006.05148. https://doi.org/10.48550/arXiv.2006.05148
Mohammad Rasouli, Tao Sun, Ram Rajagopal, (2020). FedGAN: Federated Generative Adversarial Networks for Distributed Data, https://doi.org/10.48550/arXiv.2006.07228
Yurochkin, M., Agarwal, M., Ghosh, S., Greenewald, K., Hoang, N., Khazaeni, Y., (2019). Bayesian nonparametric federated learning of neural networks, in: International Conference on Machine Learning, PMLR. pp. 7252–7261. https://doi.org/10.48550/arXiv.1905.12022
Deng, Y., Kamani, M.M., Mahdavi, M., (2020). Adaptive personalized federated learning. arXiv:2003.13461. https://doi.org/10.48550/arXiv.2003.13461
Finn, C., Abbeel, P., Levine, S., (2017). Model-agnostic meta-learning for fast adaptation of deep networks, in: International Conference on Machine Learning, PMLR. pp. 1126–1135. https://doi.org/10.48550/arXiv.1703.03400
Fallah, A.,Mokhtari, A., Ozdaglar, A.E., (2020). Personalized federated learning with theoretical guarantees: A model-agnostic meta-learning approach, in: Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual. https://api.semanticscholar.org/CorpusID:227276412
Jiang, Y., Konečn`y, J., Rush, K., Kannan, S., (2019). Improving federated learning personalization via model agnostic meta learning. arXiv:1909.12488. https://doi.org/10.48550/arXiv.1909.12488
Nichol, A., Achiam, J., Schulman, J., (2018). On first-order meta-learning algorithms. arXiv:1803.02999. https://doi.org/10.48550/arXiv.1803.02999
Chen, F., Luo,M., Dong, Z., Li, Z., He, X., (2018). Federated meta-learning with fast convergence and efficient communication. arXiv:1802.07876. https://doi.org/10.48550/arXiv.1802.07876
Hanzely, F., Richtárik, P., (2020). Federated learning of a mixture of global and local models. arXiv:2002.05516. https://doi.org/10.48550/arXiv.2002.05516
Dinh, C.T., Tran, N.H., Nguyen, T.D., (2020). Personalized Federated Learning with Moreau Envelopes, in: Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020 (NeurIPS 20). https://doi.org/10.48550/arXiv.2006.08848
Jean Jacques Moreau. Propriétés des applications " prox ". Comptes rendus hebdomadaires des séances de l'Académie des sciences, (1963), 256, pp.1069-1071. https://hal.science/hal-01867221v1
Huang, Y., Chu, L., Zhou, Z., Wang, L., Liu, J., Pei, J., Zhang, Y., (2021). Personalized cross-silo federated learning on non-iid data. Association for the Advancement of Artificial Intelligence (AAAI). https://doi.org/10.48550/arXiv.2007.03797
Mansour, Y., Mohri, M., Ro, J., Suresh, A.T., (2020). Three approaches for personalization with applications to federated learning. arXiv:2002.10619. https://doi.org/10.48550/arXiv.2002.10619
Liang, P.P., Liu, T., Ziyin, L., Allen, N.B., Auerbach, R.P., Brent, D., Salakhutdinov, R., Morency, L.P., (2020). Think Locally, Act Globally: Federated Learning

with Local and Global Representations. arXiv:2001.01523. https://doi.org/10.48550/arXiv.2001.01523

Smith, V., Chiang, C.K., Sanjabi, M., Talwalkar, A.S., (2017). Federated multi-task learning, in: Advances in Neural Information Processing Systems, pp. 4424–4434. https://doi.org/10.48550/arXiv.1705.10467

Corinzia, L., Beuret, A., Buhmann, J.M., (2019). Variational federated multi-task learning. arXiv:1906.06268. https://doi.org/10.48550/arXiv.1906.06268

Sattler, F., Müller, K.R., Samek, W., (2020). Clustered Federated Learning: Model-Agnostic Distributed Multitask Optimization Under Privacy Constraints. IEEE Transactions on Neural Networks and Learning Systems. doi: 10.1109/TNNLS.2020.3015958

Alessio Mora, Irene Tenison, Paolo Bellavista, Irina Rish, (2022). Knowledge Distillation for Federated Learning: a Practical Guide, https://doi.org/10.48550/arXiv.2211.04742

Liu, Y., Kang, Y., Xing, C., Chen, T., Yang, Q., (2020) a. Secure Federated Transfer Learning. IEEE Intelligent Systems 35, 70–82. https://doi.org/10.1109/MIS.2020.2988525

Lin, T., Kong, L., Stich, S.U., Jaggi, M., (2020). Ensemble Distillation for Robust Model Fusion in Federated Learning. 34th Conference on Neural Information Processing Systems (NeurIPS 2020) https://doi.org/10.48550/arXiv.2006.07242

Zhang, Y., Xiang, T., Hospedales, T.M., Lu, H., (2018) b. Deep mutual learning, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4320–4328. doi: 10.1109/CVPR.2018.00454

Li, C., Li, G., Varshney, P.K., 2020a. Decentralized Federated Learning via Mutual Knowledge Transfer. arXiv:(2012).13063. https://doi.org/10.1109/JIOT.2021.3078543

Hinton, G., Vinyals, O., Dean, J., (2015). Distilling the knowledge in a neural network. arXiv:1503.02531. https://doi.org/10.48550/arXiv.1503.02531

Wang, K.,Mathews, R., Kiddon, C., Eichner, H., Beaufays, F., Ramage, D., (2019)a. Federated evaluation of on-device personalization. arXiv:1910.10252. https://doi.org/10.48550/arXiv.1910.10252

Peng, X., Huang, Z., Zhu, Y., Saenko, K., (2019). Federated adversarial domain adaptation. arXiv:1911.02054. https://doi.org/10.48550/arXiv.1911.02054

Li, D., Wang, J., (2019). FedMD: Heterogenous Federated Learning via Model Distillation. arXiv:1910.03581. https://doi.org/10.48550/arXiv.1910.03581

Shoham, N., Avidor, T., Keren, A., Israel, N., Benditkis, D., Mor-Yosef, L., Zeitak, I., (2019). Overcoming Forgetting in Federated Learning on Non-IID Data. NeurIPS 2019 Workshop on Federated Learning for Data Privacy and Confidentiality. https://doi.org/10.48550/arXiv.1910.07796

Liu, B.,Wang, L., Liu,M., (2019)a. Lifelong Federated Reinforcement Learning: A Learning Architecture for Navigation in Cloud Robotic Systems. IEEE Robotics and Automation Letters 4, 4555–4562. doi: 10.1109/LRA.2019.2931179

Duchi, J., Hazan, E., Singer, Y., (2011) Adaptive Subgradient Methods for Online Learning and Stochastic Optimization. Journal of Machine Learning Research 12, 2121–2159. http://jmlr.org/papers/v12/duchi11a.html

Reddi, S., Charles, Z., Zaheer, M., Garrett, Z., Rush, K., Konečn`y, J., Kumar, S., McMahan, H.B., (2020). Adaptive federated optimization. arXiv:2003.00295. https://doi.org/10.48550/arXiv.2003.00295

Wu, Y., He, K., (2018). Group normalization, in: Proceedings of the European conference on computer vision (ECCV), pp. 3–19. https://doi.org/10.48550/arXiv.1803.08494

Ioffe, S., Szegedy, C., (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift, in: International conference on machine learning, PMLR. pp. 448–456. https://doi.org/10.48550/arXiv.1502.03167

Ghosh, A., Hong, J., Yin, D., Ramchandran, K., (2019). Robust Federated Learning in a Heterogeneous Environment. arXiv:1906.06629. https://doi.org/10.48550/arXiv.1906.06629

Ghosh, A., Chung, J., Yin, D., Ramchandran, K., (2020). An Efficient Framework for Clustered Federated Learning. arXiv:2006.04088. https://doi.org/10.48550/arXiv.2006.04088

Kopparapu, K., Lin, E., (2020) b. FedFMC: Sequential Efficient Federated Learning on Non-iid Data. CoRR abs/2006.10937, arXiv:2006.10937. https://doi.org/10.48550/arXiv.2006.10937

Luo, J., Wu, X., Luo, Y., Huang, A., Huang, Y., Liu, Y., Yang, Q., (2019). Real-World Image Datasets for Federated Learning. arXiv:1910.11089. https://doi.org/10.48550/arXiv.1910.11089

Hsu, T.M.H., Qi, H., Brown, M., (2020). Federated Visual Classification with Real-World Data Distribution, in: Proceedings of the European Conference on Computer Vision (ECCV). https://doi.org/10.48550/arXiv.2003.08082