REVIEW ARTICLE

A comprehensive survey of federated transfer learning: challenges, methods and applications

Wei GUO[1], Fuzhen ZHUANG (✉)[1,2], Xiao ZHANG (✉)[3], Yiqi TONG[1], Jin DONG[4]

1 Institute of Artificial Intelligence, Beihang University, Beijing 100191, China
2 SKLSDE, School of Computer Science, Beihang University, Beijing 100191, China
3 School of Computer Science and Technology, Shandong University, Shandong 266237, China
4 Beijing Academy of Blockchain and Edge Computing, Beijing 100080, China

Abstract Federated learning (FL) is a novel distributed machine learning paradigm that enables participants to collaboratively train a centralized model with privacy preservation by eliminating the requirement of data sharing. In practice, FL often involves multiple participants and requires the third party to aggregate global information to guide the update of the target participant. Therefore, many FL methods do not work well due to the training and test data of each participant may not be sampled from the same feature space and the same underlying distribution. Meanwhile, the differences in their local devices (system heterogeneity), the continuous influx of online data (incremental data), and labeled data scarcity may further influence the performance of these methods. To solve this problem, federated transfer learning (FTL), which integrates transfer learning (TL) into FL, has attracted the attention of numerous researchers. However, since FL enables a continuous share of knowledge among participants with each communication round while not allowing local data to be accessed by other participants, FTL faces many unique challenges that are not present in TL. In this survey, we focus on categorizing and reviewing the current progress on federated transfer learning, and outlining corresponding solutions and applications. Furthermore, the common setting of FTL scenarios, available datasets, and significant related research are summarized in this survey.

Keywords federated transfer learning, federated learning, transfer learning, survey

1 Introduction

In recent years, we have witnessed breakthroughs in machine learning, especially deep neural networks (DNNs), in various fields such as computer vision, smart cities, health care, and recommendation systems. Driven by high-quality training data, these methods have achieved impressive performance and even outperformed humans in certain tasks. With the rapid growth of the mobile Internet, a large amount of data is produced by billions of smart devices. However, these collected data cannot be directly uploaded to cloud servers or data centers for centralized processing due to limitations in data security, user privacy protection, and network bandwidth, which poses substantial challenges to the traditional machine learning approach. Such phenomena is commonly known as "isolated data islands".

One emerging paradigm for enabling distributed machine learning to solve this problem is federated learning (FL), which was first proposed by [1]. The main idea of FL is to collaboratively train a centralized machine learning model with privacy preservation by transmitting and aggregating model parameters between the distributed participants, which eliminates the requirement of local data sharing and each participant can maintain ownership of their data. However, in certain FL scenarios, the data distribution varies widely between participants. For example, the training data from different participants share the same feature space but may not share the same sample ID space, or the training data from different participants may not even share the same feature space [2]. Therefore, when participants want to utilize global information to improve model utility through FL aggregation, the difference in data distributions, feature space, and label space among participants will influence the model convergence to the optimum [3–6]. Furthermore, due to inconsistent local storage, computational, and communication capabilities among different participant devices, FL may grapple with system heterogeneity challenges, leading to straggler situations or high error rates. In addition to the above-mentioned data heterogeneity and system heterogeneity problems, FL also suffers from model heterogeneity, incremental data, and labeled data scarcity challenges, which are also focal points of attention among many researchers.

To address the aforementioned challenges, transfer learning (TL) is employed in FL as an effective method of facilitating knowledge transfer between source and target domains [7]. The main concept of TL is to minimize the divergence between the distributions of different domains. Similarly, in one communication round of FL, we could consider each

participant as the target domain and the other participants as the source domains. Given that FL often involves multiple participants, i.e., multiple source domains, and requires the central server to aggregate information (e.g., model parameters) from multiple participants to guide the update of the target participant. In the process of continuous interaction among participants, knowledge is mutually transferred, which allows a local model obtained from a specific domain to be used by other participants through TL, thus alleviating limitations such as data heterogeneity, system heterogeneity, incremental data, and labeled data scarcity. We rethink FL in [8] from the perspective of TL, and refer to the combination of FL and TL as federated transfer learning (FTL) shown in Fig. 1.

However, in classical TL strategies, the target domain can directly access the source domain data or model information, which contradicts the principle of FL. Hence, these TL strategies could not be directly applied in the FL. Moreover, the standard FL scheme contains a sending and receiving process through the communication between participant and server to ensure that the global model is updated and optimized across all local participants. So in a communication round, local participants can act as source and target domains at different stages. Concretely, during the sending stage, each participant acts as a source domain to transfer local knowledge to other participants. During the receiving stage, each participant serves as the target domain to receive knowledge from others. These conditions increase the difficulty of applying TL to FL.

Overall, the above unique challenges of FTL have captured the attention of numerous researchers, and many significant contributions have been made. Existing surveys in the FL field mainly focus on traditional FL [3,9–11], including horizontal

federated learning, vertical federated learning [6], incentive mechanism [12], privacy protection [13,14], or introducing FL applications such as healthcare [15,16], mobile edge networks [17], and internet of things (IoT) [18]. Despite some studies [19,20] focus on not identically and independently distributed (Non-IID) or other heterogeneous scenarios, such as model heterogeneity, device heterogeneity in FL, there is still a lack of systematic and comprehensive review on the definition, challenges, and corresponding solutions specific for the application of TL in FL, i.e., FTL.

To fill this gap, this survey is dedicated to giving a comprehensive survey of FTL, including definitions, a categorization, and a discussion of existing challenges and corresponding solutions, common setting scenarios of distribution heterogeneity, available datasets, as well as an outline of current FTL applications and future prospects. In detail, Fig. 2 shows the categorizations of FTL and corresponding solutions. Section 2.1 demonstrates related definitions of FL and TL, we classify the common settings of FTL scenarios into six categories, including homogeneous FTL, heterogeneous FTL, dynamic heterogeneous FTL, model adaptive FTL, semi-supervised FTL, and unsupervised FTL. Sections 3.1 to 3.5 systematically summarize the corresponding solutions of existing FTL works in these scenarios, including motivation, core algorithm, model design, privacy-preserving mechanism, and communication architecture they adopt. Since some studies have involved multiple FTL scenarios, we only describe the major issues addressed by these studies. Finally, recognizing that systems and infrastructure are critical to the success of FTL, we outline current applications of FTL and propose future prospects.

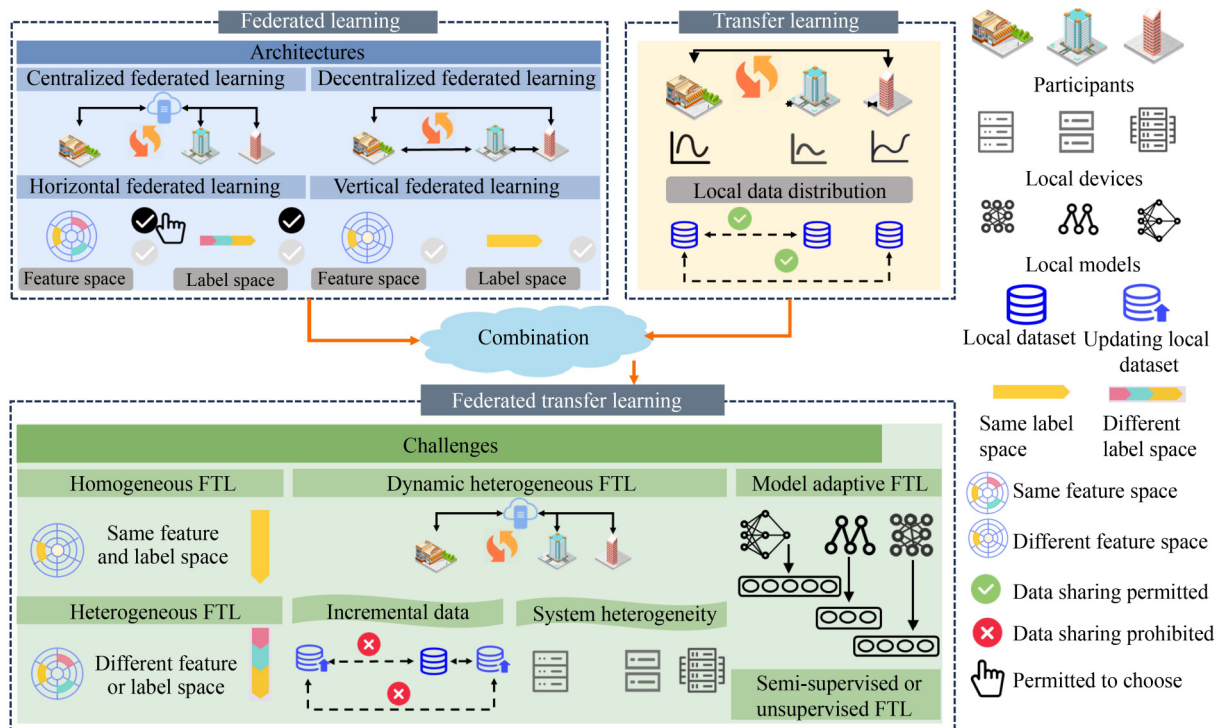The key contributions of this work are summarized as follows.
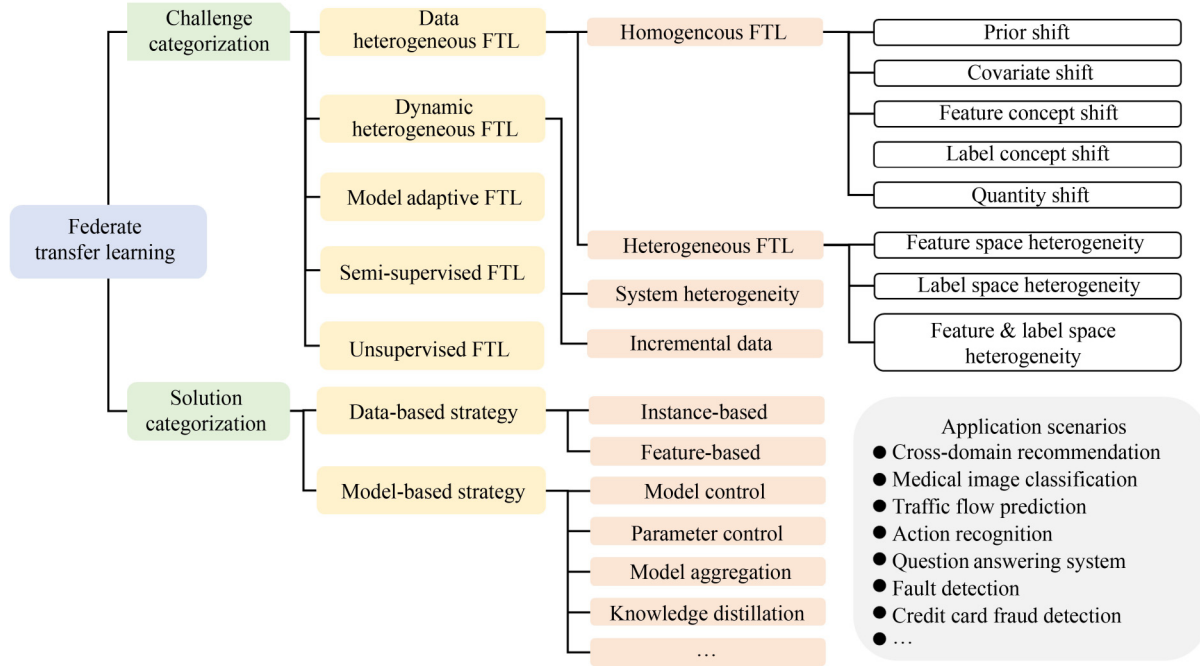

Fig. 1   The overview of FTL

Fig. 2   Categorizations of FTL

1. This survey is the first to systematically and comprehensively rethink FL based on TL (FTL). We provide the definitions of FTL and its challenges including homogeneous FTL, heterogeneous FTL, dynamic heterogeneous FTL, model adaptive FTL, semi-supervised FTL, and unsupervised FTL, and further detail these challenges of FTL through examples.
2. Based on existing FTL solutions, which include both data-based and model-based strategies, we give the current research status for FTL challenges.
3. We summarize the scenario settings of the homogeneous FTL shown in Section 2.3.1, which is the most common situation in FTL, including the setup methods

and applied datasets. Meanwhile, to make checking convenient, we outline the existing research on FTL.

2   Overview
In this section, the common notations used in this survey are listed in Table 1 for convenience. Besides, we further introduce the definitions, categorizations, and open challenges related to transfer learning, federated learning, and federated transfer learning.

2.1   Definition
Following with previous works [7,8], we first give the definitions of "domain", "task", "transfer learning", and "federated learning" that are used in this survey, respectively.

Table 1   The common notations

Symbol	Definition	Symbol	Definition
n	Number of instances	m	Number of domains
k	Number of participants	$\bar{k}$	Actual number of participants
z	Number of classes	l	Number of model layers
p	Number of servers	s	Server
u	Participant	g	Global
d	Threshold	f	Decision function
x	Feature vector	y	Label
e	Communication round	F	Device
A	Active participant	B, C	Passive participant
D	Domain	$\mathcal{T}$	Task
$\mathcal{X}$	Feature space	$\mathcal{Y}$	Label space
X	Instance space	I	Sample ID space
Y	Label set corresponding to X	S	Source domain
T	Target domain	L	Labeled instances
U	Unlabeled instances	$\mathcal{L}$	Loss function
$\mathcal{R}$	Relationship matrix	$\mathcal{M}$	Model
$\mathcal{E}$	Extractor	μ	Mean
σ	Variance	λ	Importance variable
δ	Tradeoff parameter	ρ	Interpolation coefficient
Ω	Structural risk	θ	Model parameters

The involved common notations are summarized in Table 1.

Definition 1 (Domain) A domain D is constituted by two elements: a feature space $\mathcal{X}$ and an probability distribution $P(X)$, where the symbol X represents an instance set, i.e., $X = \{x|x_i \in \mathcal{X}, i = 1,\ldots,n\}$. Thus, a domain D can be denoted as $D = \{\mathcal{X}, P(X)\}$. In general, if two domains are different, then they may have different feature spaces $\mathcal{X}$ or different probability distributions $P(X)$ [21].

Definition 2 (Task) A task $\mathcal{T}$ is constituted by a label space $\mathcal{Y}$ and a decision function f, denoted as $\mathcal{T} = \{\mathcal{Y}, f\}$. Given the training data, the decision function f is used to predict the corresponding label $y \in \mathcal{Y}$, where f is not explicit but can be inferred from the sample data.

Definition 3 (Transfer Learning) Given an/some observation(s) corresponding to $m^S \in \mathbb{N}^+$ source domain(s) and task(s) (i.e., $\{(D_{S_i}, \mathcal{T}_{S_i})|i = 1,\ldots,m^S\}$), and an/some observation(s) about $m^T \in \mathbb{N}^+$ target domain(s) and task(s) (i.e., $\{(D_{\mathcal{T}_j}, \mathcal{T}_{T_j})|j = 1,\ldots,m^T\}$), transfer learning aims to utilize the knowledge implied in the source domain(s) to improve the performance of the learned decision functions $f^{T_j}(j = 1,\ldots,m^T)$ on the target domain(s) [7].

Definition 4 (Federated Learning) Assume there are k participants $u_1,\ldots,u_k$, each aiming to train a machine learning model with their own private datasets $X_1,\ldots,X_k$. A conventional approach is to upload all data together and use $X = X_1 \cup \cdots \cup X_k$ to train a global model $\mathcal{M}_{SUM}$. However, in many application scenarios, participants cannot directly upload their own data or access the data of other participants. Therefore, a typically federated learning system is a distributed learning process in which the participant collaboratively train a model $\mathcal{M}_{FED}$ without sharing local private data X_i [8].

Definition 5 (Federated Transfer Learning) Given there are some challenges in FL for participants i and j ($i = 1,\ldots,k$), including data heterogeneity ($\mathcal{X}_i \neq \mathcal{X}_j$ or $\mathcal{Y}_i \neq \mathcal{Y}_j$ or $P_i(X_i, Y_i) \neq P_j(X_i, Y_j)$, where instance space X consists of feature space $\mathcal{X}$ and label space $\mathcal{Y}$), system heterogeneity ($F_i \neq F_j$), incremental data ($X_{e-1}^i \neq X_e^i$), and scarcity of labeled data ($\bigcup_i X_i = X^L \to 0$), the FL combines the TL to solve these challenges, called FTL. The specific definition of FTL as follows. Given k participants $u_1,\ldots,u_k$ in FL, the central server set $\{s_p|p = 1,\ldots,\mathcal{N}^+\}$ is designed to achieve model convergence over the E communication rounds.

During each communication round, the typical federated transfer learning process includes two distinct stages:

1. Sending stage: participant(s) u_i ($1 < i \leqslant k$) is(are) assumed as the role of the source domain(s) D^{S_i} ($1 < i \leqslant m^S$), where they are responsible for contributing local an/some observation(s) (D_{S_i}, $\mathcal{T}_{S_i}$) corresponding to D_{S_i} and task(s) $\mathcal{T}_{S_i}$ to the central server s_p without sharing local raw data $X_1,\ldots,X_k$. The server then leverages the collected sending information to implement the aggregation process.

2. Receiving stage: once the aggregation is complete, participant(s) u_j ($1 < j \leqslant k$) then are assumed as the role of the target domain(s) $\{(D_{\mathcal{T}_j}, \mathcal{T}_{T_j})|j = 1,\ldots,m^T\}$ and utilize the received global aggregation information to perform local model updates.

The above sending and receiving stages are assumed to repeat for E communication rounds, or until the model is observed to converge. Particularly, when $p = 0$, the above process is considered as a decentralized federated transfer learning process.

2.2 Category of federated learning

According to the characteristics of data distribution among connected participants, FL can be categorized into horizontal FL (HFL) and vertical FL (VFL). Generally, HFL considers the distributed participants to have data with the same features but are different in sample space, while VFL considers the distributed participants to have the same samples but different features to jointly train a global model [6,19]. Federated transfer learning in [8] refers that these participants have differences in both feature space and label space. Due to the limited research on federated transfer learning in [8], this survey categorizes federated transfer learning and VFL as a type of VFL for description.

On the other hand, depending on whether there is a/some central server(s) responsible for coordinating participants, FL can also be divided into centralized FL (CFL) and decentralized FL (DFL), where CFL assumes that there is a/some server(s) to gather local model-related information or other training information from the participants and then distributes the updated global model back to the participants, while DFL assumes participants directly aggregate information from neighboring participants [8]. In the following, we will provide a brief introduction to these FL frameworks and discuss the various settings of source domains, target domains, and tasks when employing transfer learning within these frameworks.

2.2.1 Horizontal federated learning

HFL is commonly found in scenarios where participants share the same feature space $\mathcal{X}$ but different sample space I, which meets homogeneity FTL described in Section 2.3.1. For example, the medical record data of two regional hospitals i and j may be very similar due to they use the same information system, which both record the patient's name, age, gender, and other user private data, so their feature spaces are the same ($\mathcal{X}_i = \mathcal{X}_j$). However, the two hospitals have different user groups ($I_i = I_j$) from their respective regions, and the user intersection of their local datasets is very limited. In FTL, any participant can serve as a source domain (S) to provide knowledge or as a target domain (T) to receive knowledge from other participants in the same feature space, therefore, we define HFL in homogeneous FTL as:

$$\mathcal{X}_i^S = \mathcal{X}_j^T, \mathcal{Y}_i^S = \mathcal{Y}_j^T, I_i^S \neq I_j^T, \forall X_i^S, X_j^T$$

$$\text{or } \mathcal{X}_i^T = \mathcal{X}_j^S, \mathcal{Y}_i^T = \mathcal{Y}_j^S, I_i^T \neq I_j^S, \forall X_i^T, X_j^S, i \neq j.$$

From an extended perspective, HFL meets heterogeneous FTL

when participants' label space is inconsistent in the knowledge-transferring process, the HFL in heterogeneous FTL can be represented as:

$$\mathcal{X}_i^S = \mathcal{X}_j^T, \mathcal{Y}_i^S \neq \mathcal{Y}_j^T, I_i^S \neq I_j^T, \forall X_i^S, X_j^T$$

$$\text{or } \mathcal{X}_i^T = \mathcal{X}_j^S, \mathcal{Y}_i^T \neq \mathcal{Y}_j^S, I_i^T \neq I_j^S, \forall X_i^T, X_j^S, i \neq j.$$

2.2.2   Vertical federated learning

Unlike HFL where all participants have their own local data labels, in the VFL scenario, participants' feature spaces $\mathcal{X}$ are inconsistent, and their sample spaces I may also not be entirely the same. For example, suppose there is a high degree of overlap in the customer groups I between a bank i and a telecommunications company j in the same region. The bank i has information on users' credit history ($\mathcal{X}_i$), such as loan repayment details and credit card usage, while the telecommunications company holds data on users' call logs, data usage, and payment records ($\mathcal{X}_j$), where the feature space is different ($\mathcal{X}_i \neq \mathcal{X}_j$). These two entities, which all act both as source domains or as target domains, can engage in VFL to mutually enhance their services in different feature spaces. We summarize VFL in heterogeneous FTL as:

$$\mathcal{X}_i^S \neq \mathcal{X}_j^T, \mathcal{Y}_i^S = (\neq)\mathcal{Y}_j^T, I_i^S = (\neq)I_j^T, \forall X_i^S, X_j^T$$

$$\text{or } \mathcal{X}_i^T \neq \mathcal{X}_j^S, \mathcal{Y}_i^T = (\neq)\mathcal{Y}_j^S, I_i^T = (\neq)I_j^S, \forall X_i^T, X_j^S, i \neq j.$$

2.2.3   Centralized federated learning

Standard CFL requires one or more central servers to build a global model by collecting local information from distributed participants [22], which involves three fundamental steps as described below:

1. Receiving stage: each participant receives the initial model sent by the server.
2. Sending stage: participants u_i use their own private data X_i to train the local model (add local model notion), and then send the local model to the server.
3. Receive stage: The central server updates the global model $\mathcal{M}_g$ by collecting and aggregating all the local updates and then sends the updated global model back to the participants.

In the FTL setting, during the sending and receiving stages, participants share knowledge through a central aggregation strategy, where each participant can act as a source domain providing knowledge or a target domain receiving knowledge. For instance, during the sending stage, participants act as source domains providing model parameters, while the server acts as the target domain, aggregating these parameters to form a global model. Conversely, in the receiving stage, the server serves as the source domain providing global model parameters to each participant.

2.2.4   Decentralized federated learning

Compared with CFL, DFL is conducted over different participants $\{u_1, \ldots, u_k\}$ without a central parameter server for global model aggregation. Each participant uses a private local dataset to optimize their local model after receiving model updates from other participants. This process involves two fundamental steps as described below:

1. Receiving stage: participant u_i federally train its initial model $\mathcal{M}_i$ locally with its own dataset X_i, and then send the model $\mathcal{M}_i$ to other participants $\{u_1, \ldots, u_{k-1}\}$ without direct data exposure.
2. Sending stage: participant u_i obtain the aggregated model $\mathcal{M}_g$ by aggregating the received local model $\{\mathcal{M}_1, \ldots, \mathcal{M}_k\}$, and then update local model with aggregated model.

Similar to CFL, each participant in DFL could still serve as either a source domain or a target domain without a central server during different stages of the FL process.

2.3   Federated transfer learning

Transfer learning has achieved remarkable success by enabling the application of knowledge from one domain to improve performance in another, significantly reducing the need for extensive data collection and training time in new tasks [23–26]. However, as shown in Fig. 3, constrained by the unique distributed learning paradigm of FL, current FTL studies face many additional challenging situations, including homogeneous FTL, heterogeneous FTL, dynamic heterogeneity FTL, model adaptive FTL, semi-supervised FTL and unsupervised FTL. More descriptions are presented below.

2.3.1   Homogeneous federated transfer learning

Assume that the local data of participant i and participant j constitute a source domain D_i and a target domain D_j, respectively. $D_i \neq D_j$ represents a difference in either their feature spaces ($\mathcal{X}_i \neq \mathcal{X}_j$) or marginal distributions ($P_i(X)$ or $P_j(Y)$). Similarly, if the task between participant i and participant j is not same, that is $\mathcal{T}_i \neq \mathcal{T}_j$, then there is a difference in their label spaces ($\mathcal{Y}_i \neq \mathcal{Y}_j$) or in their conditional distribution ($P_i(y|x) \neq P_j(y|x)$).

HOFTL refers to differences in marginal distributions ($P_i(X)$ or $P_j(Y)$), conditional distributions ($P_i(y|x) \neq P_j(y|x)$), or sample sizes $n_i \neq n_j$ between participant data, which is often caused by diversity in domain or task between participants. Based on this, HOFTL includes five scenarios:

● Prior shift: $P_i(Y) \neq P_j(Y)$
● Covariate shift: $P_i(X) \neq P_j(X)$
● Feature concept shift: $P_i(x|y) \neq P_j(x|y)$
● Label concept shift: $P_i(y|x) \neq P_j(y|x)$
● Quantity shift: $n_i \neq n_j$

Specifically, as described in Subsection 2.2.1, horizontal federated learning assumes that participants have the same feature space, so HOFTL is a form of transfer learning under this HFL assumption. Moreover, HOFTL can also be presented in vertical federated learning when there is partial overlap in the feature space between participants. Unless specifically stated, the HOFTL methods discussed in this survey are all related to HFL. Overall, compared with homogeneous transfer in traditional transfer learning that only considers marginal and conditional probability distributions,
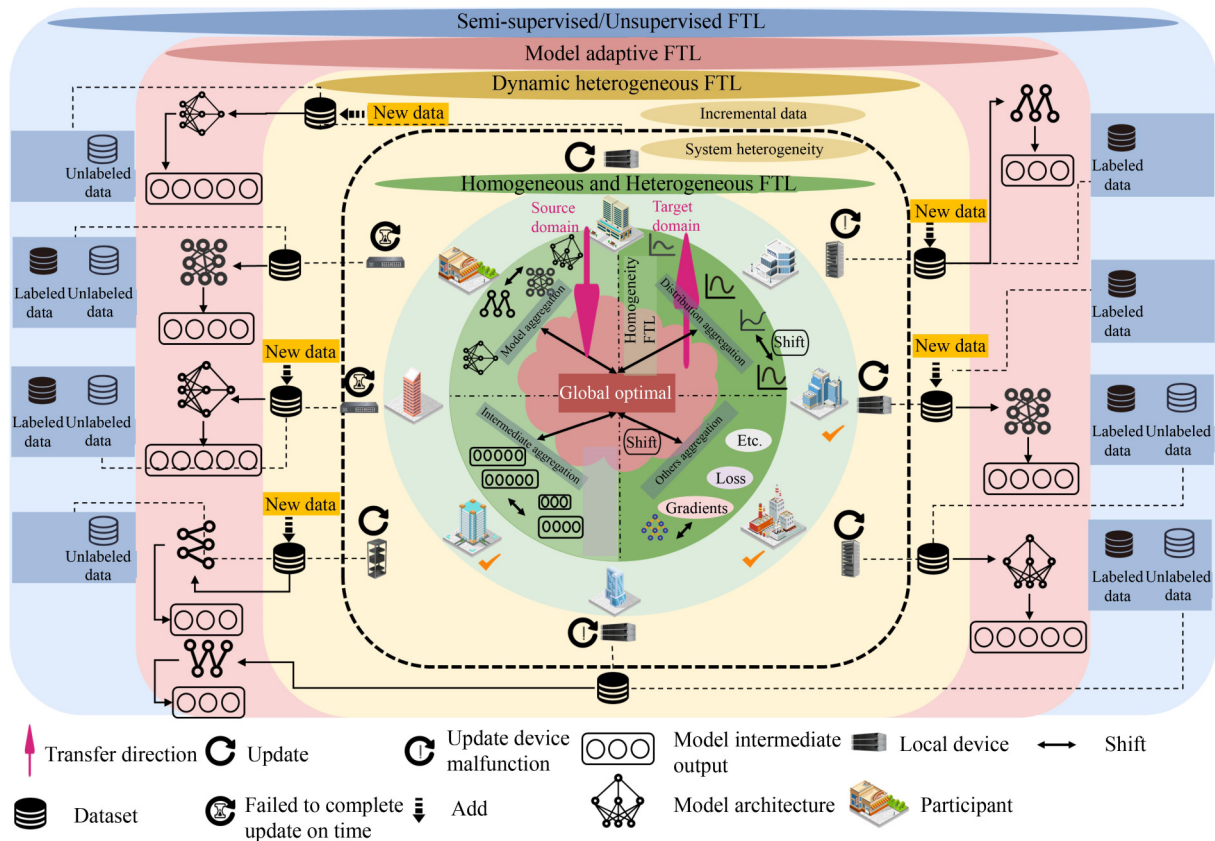
Fig. 3 The challenges of FTL

HOFTL also considers changes in the total sample size, which corresponds to the Non-IID data setting in federated learning. The detailed challenges of HOFTL are described below, and considering that homogeneous FTL is one of the most frequently discussed, we have summarized the specific settings for each homogenous FTL scenario demonstrated in Table 2.

● Prior shift
Prior shift, also known as class imbalance, implies that the prior probability distribution $P(Y)$ could be inconsistent between different participants when the conditional probability distribution $P(y|x)$ is consistent [182]. In FL, the prior probability distribution inconsistency may occur when different participants have different class distributions in their local datasets. If these differences are not properly handled, they can lead to a federated model that performs unfair and suboptimal performance. For example, an FL system is designed to improve predictions for a specific disease (e.g., diabetes) across different hospitals that participate in model training without sharing private patient records. Hospital A is located in an urban area with a high prevalence of diabetes, possibly due to lifestyle factors prevalent in the population it serves. As a result, in hospital A's patient data, 30% of patients might have diabetes. On the other hand, hospital B serves a rural area with a different demographic and lifestyle, resulting in only 10% of its patients having diabetes. This difference in the prevalence of diabetes is a classic example of prior probability shift. In some extreme cases, hospital B may even have no positive cases for this disease.

● Covariate shift
Covariate shift, or feature distribution imbalance, describes a situation where the input feature distribution $P(X)$ is varied between participants while the conditional probability $P(y|x)$ remains consistent. This presents a unique challenge in FL because the global model is trained on data from multiple participants, and each participant's local data may represent a different underlying distribution of input features. For example, the patient population at hospital A had a higher average body mass index (BMI), which is a known risk factor for diabetes, while the patient population at hospital B had a lower average BMI. There is a significant difference in the input data (in this case, the BMI distribution) between the two hospitals, known as covariate shift.

● Feature concept shift
Concept drift, which includes feature concept shift and label concept shift, refers to the change in the relationship between variables x and y, where feature concept shift implies to $P(x|y)$ discrepancy among participants with the same prior distribution $P(y)$ [182]. This type of shift can be particularly challenging in federated learning because models need to generalize across all participants' data. For example, consider two hospitals A and B jointly predicting the incidence of diabetes, where "x" represents the patient's health characteristics and "y" represents the presence or absence of diabetes. Hospital A's diabetic population mostly has a higher socioeconomic status, resulting in a different set of health characteristics, such as better control of blood sugar levels and fewer complications. In contrast, patients with diabetes at

Table 2   Data heterogeneity settings of HOFTL and HEFTL

Problem categorization	Setting		Reference	Dataset
HOFTL	Prior shift	Fixed ratio	[27–33]	CIFAR-10[1], CIFAR-100[1], MNIST[2], Tiny-Imagenet[3], ImageNet[3], FEMNIST[4], OFFICE [34], DIGIT [35], OpenImage [36], WESAD [37], KDD99[5], SVHN[6], HAR[7], OFFICE-Caltech 10[8], MIMIC-III[9], Shakespeare [1], DomainNet[10], NSL-KDD99[11], CINIC10 [38], CelebA [39], StackOverflow [40]
		Natural partition	[41–57]	
		1 class/participant	[37,58–60]	
		> 1 classes/participant	[1,48,59,61–117]	
		Dirichlet Distribution	[118–121,121–133], [32,81,93,112,134–146]	
		JensenShannon divergence	[147]	
		Half-normal distribution	[47,148]	
		Log-normal distribution	[79]	
	Covariate shift	1 domain/participant	[41,65,125,149–151], [1,70,76,129,152–156], [89,91,96,112,157–166]	
		Mixed domain/participant	[167,168]	
	Feature concept shift	1 degree/participant	[169]	
	Label concept shift		[89,163]	
	Quantity shift	Natural	[52,54,89,98,159,170,171], [109,114,161,172]	
		By data source	[1,61,125,157]	
		By parameter	[88]	
HEFTL	Feature space hetergeneity	Overlapped feature	[37,87,133,173–175]	CIFAR-10[1], CIFAR-100[1], MNIST[2], MovieLens [176], ModelNet [177], FEMNIST[4] NUS-WIDE [176]
		Non-overlapped feature	[176,178–181]	
	Label space heterogeneity			
	Feature and label space heterogeneity			

hospital B may have lower socioeconomic status and poorer health characteristics, such as uncontrolled blood sugar levels and higher rates of complications. Differences in the distribution of health characteristics (x) for a given diabetic patient (y) are an example of a $P(x|y)$ shift.

● Label concept shift
Similar to feature concept shift, the label concept drift refers to $P(y|x)$ inconsistent among participants with the same covariate distribution $P(x)$ [182]. Some external events or changes may lead to changes of $P(y|x)$ in either the source/target domain, which further renders the models from the source/target domain no longer suitable for tasks in the target/source domain. For example, in a federated recommendation system, geographical location is commonly used as the input feature to predict users' favorite items. Thus, if the emergence of tendentious policy or new pillar industries supports the economic development of area A, the consumption level of A will be improved. In this situation, the expected user preference will change, causing the prediction results of participant u_i from A to become invalid and unsuitable for the improvement of model predictions from other regions'

participants.

● Quantity shift
Different from the prior shift, quantity shift refers to the situation where there is a significant imbalance in the number of training samples available among participants. In FL, some participants might have a large dataset, while others may have a relatively small one. This can lead to a situation where the global model is disproportionately influenced by participants with more data, potentially leading to biases or overfitting to the characteristics of those datasets. For example, a large-scale hospital may have thousands of patient records, while a small clinic may only have a few hundred. This difference in data volume is a classic example of quantity shift in FL.

In summary, with uniform feature and label spaces, participants in homogeneous FTL still face data distribution shift problems, including prior shift, covariate shift, feature concept shift, label concept shift, and quantity shift. Most current FTL studies focus on prior and quantity shifts, with few studies tackling covariate shifts. Feature concept shift and label concept shift are even less explored. However, external elements change, like time or policy, may change the

---

1) See cs.toronto.edu/~kriz/cifar.html website.

2) See kaggle.com/datasets/hojjatk/mnist-dataset website.

3) See kaggle.com/c/tiny-imagenet website.

4) See github.com/wenzhu23333/Federated-Learning website.

5) See kdd.ics.uci.edu/databases/kddcup99 website.

6) See ufldl.stanford.edu/housenumbers website.

7) See github.com/xmouyang/FL-Datasets-for-HAR website.

8) See v7labs.com/open-datasets/office-caltech-10 website.

9) See physionet.org/content/mimiciii-demo/1.4 website.

10) See ai.bu.edu/M3SDA website.

11) See s.uci.edu/dataset/227/nomao website.

relationship between features and labels for partial participants while leaving others unchanged. This intensifies the feature concept drift and label concept drift among participants, which is worth deeper study in the future.

2.3.2 Heterogeneous federated transfer learning

HEFTL mainly refers to the problem of inconsistency in feature or label space between participants in FL. To the specific, similar to HOFTL, it is assumed that there are two participants i and j, whose private data constitute the source domain D_i and the target domain D_j, respectively. HEFTL demonstrates the differences in either their domain ($D_i \neq D_j$) or task ($\mathcal{T}_i \neq \mathcal{T}_j$), which caused by their various feature spaces ($\mathcal{X}_i \neq \mathcal{X}_j$) or/and label space ($\mathcal{Y}_i \neq \mathcal{Y}_j$). Based on this, HEFTL has three scenarios:

- Feature space heterogeneity: $\mathcal{X}_i \neq \mathcal{X}_j$
- Label space heterogeneity: $\mathcal{Y}_i \neq \mathcal{Y}_j$
- Feature and label space heterogeneity: $\mathcal{X}_i \neq \mathcal{X}_j$ and $\mathcal{Y}_i \neq \mathcal{Y}_j$

Unlike HOFTL, where all participants train models with the same data structure, HEFTL allows for collaboration between datasets that are not identically structured. Thus, vertical federated learning is a prime case for HEFTL, since the local private data among participants in VFL may contain different sets of attributes or dimensions. Next, we give a detailed description of the settings as mentioned above.

● Feature space heterogeneity

Feature space heterogeneity refers to the situation that the feature space $\mathcal{X}$ of different participants is inconsistent, while the label space $\mathcal{Y}$ is consistent, particularly when different datasets involved in the training process have different sets of features. For example, in FL, two retailers are trying to identify fake reviews by local model prediction. Retailer A has a feature space that includes review length, the number of purchases, and purchase history, whereas retailer B utilizes review timing, user location, and account age as feature space. They all annotated their reviews with binary labels as "true" (0) or "fake" (1). Although the reviews obtained by different retailers have inconsistent feature space, these retailers still aim to leverage FL to enhance the predictive performance of their respective local models within a consistent label space.

● Label space heterogeneity

Label space heterogeneity refers to the situation where different participants have consistent feature space $\mathcal{X}$ but inconsistent label space $\mathcal{Y}$, which is the exact opposite of feature space heterogeneity. For example, in FL, two international e-commerce platforms are aiming to improve their recommendation systems. Each platform operates in a different region and thus has different product categories that are relevant to their local markets. Platform A serves the Asian market and uses categories like "Apparel", "Gadgets", "Furniture", and "Anime Merchandise". Platform C is based in North America, and uses labels such as "Clothing", "Tech", "Home Improvement", and "Sports Equipment". All two platforms collect user data including browsing time, click-through rates, purchase history, and search queries, which make up their consistent feature space. However, the way they categorize their products (labels) varies due to regional differences in terminology and market demand, leading to an inconsistent label space.

● Feature and label space heterogeneity

Feature and label space heterogeneity, indicates the feature space $\mathcal{X}$ and label space $\mathcal{Y}$ are both inconsistent among different participants. For example, there are two different specialty health clinics using federated learning to predict if patients will need to return for more treatment. Each clinic has its own set of measurements and outcomes. Clinic A focuses on heart health, measuring things like heartbeat patterns and blood tests, and is concerned with whether patients might come back with heart issues. Clinic B is a general clinic in a remote area, tracking health indicators like blood pressure and weight, and wants to predict if patients will return for any follow-up care or need a specialist. Each clinic collects different health information (different feature spaces) and has different categories for what counts as a patient needing to return (different label spaces). Considering the health indicators may be helpful in predicting patient's heart issues in the future. Thus, they want to use FL to build better prediction models without sharing sensitive patient data.

In summary, heterogeneous FTL may occur when there is inconsistency in the participants' feature spaces or label spaces. The existing research primarily focuses on FTL with heterogeneous feature spaces where only the feature spaces are inconsistent. Other heterogeneous situations in FTL remain worthy of deeper investigation.

2.4 Dynamic heterogeneous FTL

DHFTL refers to the condition where the participant set that contributes to the FTL aggregation or the local raw data of partial participants in this set is dynamically changing at each round. We further provide detailed descriptions of the causes of dynamic heterogeneity.

2.4.1 System heterogeneity

Each participant's local device F in FL could have different storage, computation, and communication abilities. Due to the varying storage or computational capabilities, some devices may not be able to complete the local training in time before aggregation. Meanwhile, the communication ability among participants is also influenced by network connections, and some devices may lose connection during a communication round because of connectivity or power issues [70,183,184]. These aspects greatly amplify the straggler issue in the aggregation process [185], forming a dynamically changing set of participants during FL iterations as shown in Fig. 4. Assuming that there is a global optimal direction r_{e-1}^g for the global model aggregated by participants $u_1, ..., u_n$ in communication round e, and $m (0 \leqslant m \leqslant k, n \neq m)$ participants could send their local model to server in time due to device's limitation in communication round e, the global optimal direction r_e^g aggregated by participants in round e may have a significant difference with r_{e-1}^g when there is data heterogeneity among participants, which is not conducive to
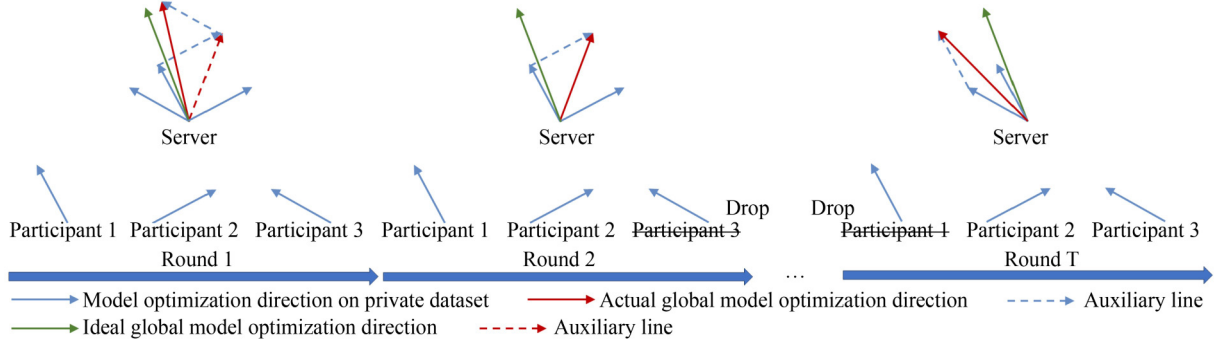
Fig. 4 The detailed description of system heterogeneity in FTL. In each round of global communication, due to the resource heterogeneity among the participants, partial participants could not participate in the global aggregation in time, which results in the actual optimization direction of the aggregated model dynamically changing and deviating from the global optimal optimization direction

the global model convergence. Therefore, how to transfer the knowledge among participants within a dynamically changing participant set is a key challenge for DHFTL. Dynamic heterogeneous FTL for participants $u_1, \ldots, u_k$ caused by system heterogeneity ($F_i \neq F_j, i, j \in (1, k)$) can be expressed as:

$$D_1^S, \ldots, D_n^S = \mathbb{S}_{e-1} \neq \mathbb{S}_e = D_1^S, \ldots, D_m^S,$$

where n and m represent the actual number of participants in communication round $e-1$ and e of FTL, respectively. $\mathbb{S}_e$ indicates the set of actual participants in the communication round e.

2.4.2   Incremental data

Real-world FL applications are often dynamic, where local participants receive the new data, classes or tasks in an online manner [102,186,187], which proposes a key challenge is how to execute FTL from dynamically changing data distributions [188,189]. If only some participants are constantly adding data, or even if each participant synchronously adds new data, the newly added data could disrupt the original local data distribution, potentially exacerbating the differences between participant distributions as represented in Fig. 5. This requires the model to generalize well across both the old and new domains [190,191]. In addition, it's also possible that the feature space of the newly added data is inconsistent with the original feature space. Dynamic heterogeneous FTL for participants $u_1, \ldots, u_k$ caused by incremental data ($X_{e-1}^i \neq X_e^i$

or $\{\mathcal{X}, \mathcal{Y}\}_{e-1}^i \neq \{\mathcal{X}, \mathcal{Y}\}_e^i, i \in (1, k)$) in communication round e. Dynamic heterogeneous FTL in a participant can be written as:

$$P(X_{i,e-1}^S) \neq P(X_{i,e}^S) \, or \, \{\mathcal{X}, \mathcal{Y}\}_{i,e-1}^S \neq \{\mathcal{X}, \mathcal{Y}\}_{i,e}^S.$$

Another situation where dynamic heterogeneous FTL of multiple participants can be denoted as:

$$P(X_{i,e}^S) \neq P(X_{j,e}^S) \, or \, \{\mathcal{X}, \mathcal{Y}\}_{i,e}^S \neq \{\mathcal{X}, \mathcal{Y}\}_{j,e}^S.(i, j \in (1, k)).$$

Nevertheless, we can only observe popularity in typical incremental learning approach [188,189,191,192], while these problems in incremental FTL receive relatively less attention.

2.4.3   Model adaptive FTL

In practical scenarios, due to differences in training objectives, participants may employ different model architectures $\mathcal{M}$ for training [164,193]. Therefore, employing conventional aggregation methods for heterogeneous model's output representations or parameters, such as averaging participants' parameters in FedAvg [1], cannot effectively complete knowledge transfer between participants in FL. Additionally, even if the dimensions of the intermediate feature outputs are consistent across different models, the representational capacity of these features for local data could still vary, hindering performance improvement of the model in the target participant [66,164,193,194]. TL generally assumes that the model architectures in the source domain and the target domain are consistent, however, the inconsistency of models
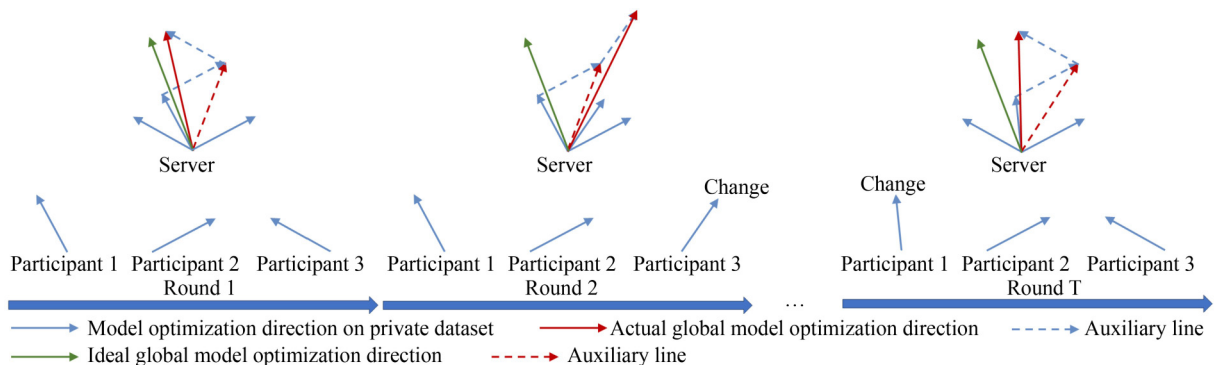

Fig. 5 The detailed description of incremental data in FTL. In each round of global communication, due to the increase in the local user data or class, the local data distribution of participants may change, causing the actual optimization direction of the aggregated model to constantly vary and deviate from the global optimal optimization direction

in FL poses a new challenge for the performance of target participants' tasks by aggregating process. This challenge, implementing effectively federated training in a model heterogeneous setting, is referred to as model adaptive FTL, which can be denoted as for participants $u_1, \ldots, u_k$: $\mathcal{M}_i \neq \mathcal{M}_j, i, j \in (1, k)$

2.4.4 Semi-supervised and unsupervised FTL

Real-world FL applications especially need to use unlabeled data more than others [195,196]. On the one hand, in cross-device federated learning [182], individual devices create a lot of unlabeled data, like photos, texts, and health record data from wearables. It's unrealistic to label all this data due to its large volume. On the other hand, cross-silo FL [182] involves businesses, where data labeling often needs special knowledge. This is common in finance and healthcare sectors. Labeling this data would be time-consuming and expensive. Thus, SSFTL and USFTL have caught the interest of some researchers [122,123,195]. Overall, SSFTL has two common scenarios: ① only one participant has labeled data; ② several participants each have a small amount of labeled data locally, where case ③ is often seen in VFL, where it's usually assumed that only one active party has data label information. USFTL in FTL refers to the scenario where all participants lack labeled information. The labeled data scarcity in FTL for participants $u_1, \ldots, u_k$ is denoted as: $\bigcup_i X_i = X^L \to 0, i \in (1, k)$.

3 Methodology

We elaborate on the current research strategies for each FTL challenge mentioned in Section 2.3. As shown in Fig. 6, it mainly includes two mainstreams: data-based and model-based strategies. Specifically, data-based strategies emphasize knowledge transfer by modulating and transforming participants' data for space adaptation, distribution adaptation, and data attribute preservation or adjustment [7] without exposing any raw private data. The model-based strategies aim to improve the predictive accuracy of any given participant by the models from other participants in FTL. Tables 3–5 demonstrate related works on solving FTL challenges through

these strategies. Note that currently there are very few FL works that specifically address the issues of label space heterogeneity or label & feature space heterogeneity. Therefore, we will not discuss them in a separate subsection.

3.1 Homogeneous federated transfer learning

Homogeneous FTL and heterogeneous FTL are two of the most studied challenges in FTL. We will first illustrate the strategies for addressing homogeneous FTL challenges from both data-based and model-based perspectives as shown in Fig. 6.

3.1.1 Prior shift

This subsection describes solutions to the prior shift challenge in HOFTL, which is one of the most common issue in FTL.

● Instance augmentation

Instance augmentation in FTL aims to enhance data homogeneity of various participants through techniques like oversampling [27,28,208–210] and undersampling [211–213], which mainly occurs on the side of the participants. In detail, FedHome [28], Astraea [47], and FEDMIX [61] consider resolving prior probability bias at the local participant level. However, they ignore the effectiveness of global information in the local augmentation process. Therefore, some studies [118,119] suggest bridging the gap between participant and global distribution by creating a public dataset, but this also increases the risk of privacy leakage. To mitigate this issue, Faug [29], an FTL approach based on the generative adversarial network (GAN), is proposed to avoid privacy issues from multiple data transfers. Faug trains a GAN on minority class data at a central location, then sends it back to participants for data generation, helping build independent and identically distributed (IID) datasets. However, the construction of the generator increases extra computational and communication costs. the study [62] introduces a batch normalization (BN) based data augmentation approach, involving the following steps:

1. BN layer parameterization: In the t round of global iteration, the ith BN layer ($i \in 1, \ldots, l$) of the global
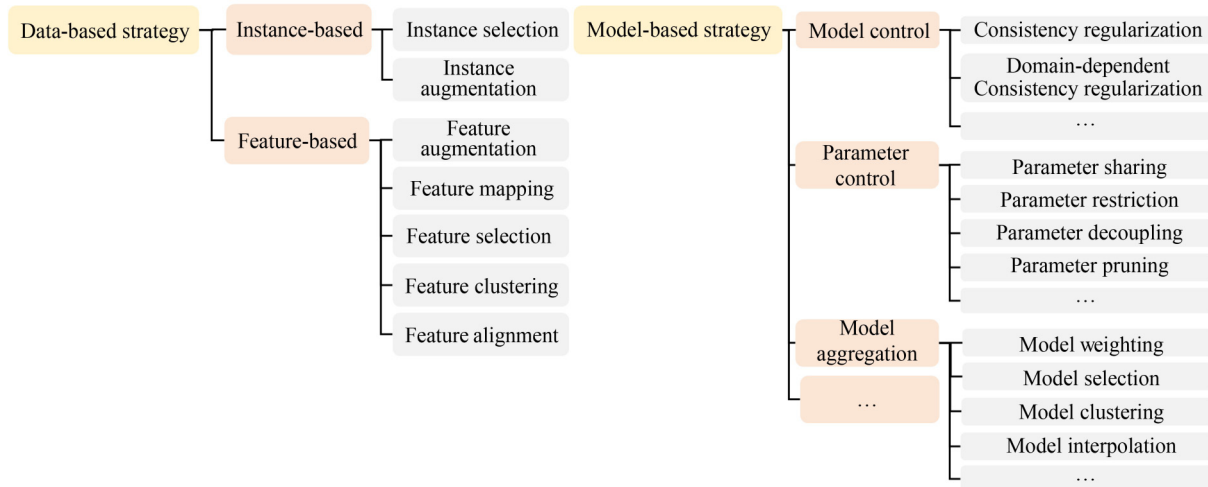

Fig. 6 Data-based and model-based strategies of FTL

Table 3   FTL frameworks

Reference	DHFTL		MAFTL	SSFTL	USFTL	HOFTL					HEFTL			HFL	VFL	CFL	DFL	Strategy
	System	Incremental				PS	CS	FCS	LCS	QS	FSH	LSH	FLSH					
[27,28,62]						√								√		√		IA
[47]						√								√		√		IA,MS,FC
[120]				√		√								√		√		IA
[61]						√			√					√		√		IA
[29]						√								√		√		IA, KD
[121]				√		√								√		√		IA, FC
[30,58]						√								√		√		IS
[197]	√					√								√		√		IS
[198]	√								√					√		√		IS
[123]				√		√								√		√		IS
[149]							√							√		√		FA
[176]			√	√							√				√		√	FA
[172]									√		√				√		√	FA
[125]						√	√		√					√		√		FA
[126]						√								√		√		FM
[199]					√		√							√		√		FM
[200]			√											√		√		FM
[173]				√							√				√		√	FC
[167]							√							√		√		FC,CR
[37,175]											√			√		√		FS
[174,178]											√				√	√		FS
[49]						√								√			√	FS
[181]				√							√				√	√		FS
[179]				√							√				√		√	FS
[65]						√	√							√		√		FC,MC
[48]					√	√								√		√		FS
[180]				√							√				√	√		FS,MS
[150]							√		√					√		√		FAI
[103]			√			√								√		√		FAI
[201]			√		√									√		√		FAI
[66]			√			√			√					√		√		FAI,MC
[41]				√		√								√		√		CR
[151]							√							√		√		CR
[102]			√			√								√		√		CR,MS
[50]				√		√								√		√		CR,PD
[72]						√								√		√		CR,PR
[67]				√		√								√		√		DCR,KD
[128]						√								√		√		DCR
[153]							√							√		√		DCR

[1] Abbreviation: PS: prior shift; CS: covariate shift; FCS: feature concept shift; LCS: label concept shift; QS: quantity shift; FSH: feature space heterogeneity; LSH: label space heterogeneity; FLSH: feature and label space heterogeneity; HFL: horizontal FL; VFL: vertical federated learning; CFL: centralized FL; DFL: decentralized FL; IA: instance augmentation; IS: instance selection; FA: feature augmentation; FM: feature mapping; FS: feature selection; FC: feature clustering; FAI: feature alignment; CR: consistency regularization; DCR: domain-dependent consistency regularization; PS: parameter sharing; PR: parameter restriction; PD: parameter decoupling; PP: parameter pruning; MW: model weighting; MS: model selection; MC: model clustering; MI: model interpolation; KD: knowledge distillation.

model $\mathcal{M}_g$ from the $(i-1)$th round can be parameterized as a distribution of means μ_i and variances σ_i. For a given target category $\bar{y}(j)(1 \leqslant j \leqslant z)$, where z represents the total number of possible categories, participants will sample from the Gaussian distribution to generate samples $\bar{x}(z)$ by forward propagate $\mathcal{M}(\bar{x}(j))$ $(1 \leqslant j \leqslant z)$, meanwhile the intermediate activation values a_i produced in sample process are applied to obtain the BN statistics $\bar{\mu}_i, \bar{\sigma}_i$).

2. Augmented Data Update: The computation of the loss function follows the formula: $\bar{x}(j) = argmin_{\bar{x}} \sum_{i=1}^{l} \|\bar{\mu}_i - \mu_i\|_2^2 + \|\bar{\sigma}_i - \sigma_i\|_2^2 + \mathcal{L}_\mathcal{H}(\mathcal{M}$ $(\bar{x}(j)), \bar{y}(j))$, where $\mathcal{H}$ represents the cross-entropy loss. During the backpropagation process, the parameters of the model $\mathcal{M}$ are kept fixed, and only $\bar{x}(j)$ is updated to obtain augmented data that is closer to the real distribution.

● Instance selection

Instance selection aims to select a subset of the available data that is most representative or informative for the training process. In distributed learning, numerous studies [214–220] suggest that constructing local training samples that are closer to the target distribution through instance selection methods can effectively enhance model performance. However, these methods are designed for traditional distributed training where training data is public. They are not suitable for FL, which uses private datasets from different owners. To solve this

Table 4 FTL frameworks (continued)

Reference	DHFTL		MAFTL	SSFTL	USFTL	HOFTL					HEFTL			HFL	VFL	CFL	DFL	Strategy
	System	Incremental				PS	CS	FCS	LCS	QS	FSH	LSH	FLSH					
[69]						√								√		√		PS
[154]							√							√		√		PS
[68,117]						√								√		√		PS,PD
[202]	√					√	√							√		√		PR
[75,77,78,130,131]						√								√		√		PR
[74,76]						√	√							√		√		PR
[73]						√								√		√		PR,MI
[71]						√	√							√		√		PR,MI
[63,79,80,83]						√								√		√		PD
[155]							√							√		√		PD
[81]				√		√								√		√		PD
[84]				√		√								√		√		PD
[82]					√	√								√		√		PD
[156]	√						√							√		√		PD
[110]		√											√		√	√		PD,MI
[106]	√					√								√		√		PD,MC
[133]				√		√	√			√				√	√	√		PD,KD
[132]						√								√		√		PD,PP,MS
[129]						√	√							√		√		PD,PR,MI
[87]				√		√				√				√	√	√		PP
[85]	√					√								√		√		PP
[86]	√					√								√		√		PP,MC
[134]	√					√								√		√		PP,KD
[1]	√					√				√				√		√		MW
[157]					√		√			√				√		√		MW
[88]						√				√				√		√		MW
[158]							√							√		√		MW
[203]						√								√		√		MW
[204]							√							√		√		MW
[90]				√		√								√		√		MW
[135]						√								√		√		MW,CR
[89]						√	√		√	√				√		√		MW,CR
[152]				√			√							√		√		MW,PD
[122]					√	√								√		√		MW,MC
[91]					√		√							√			√	MS
[60,92,93,136,148]						√								√		√		MS
[51]					√	√								√			√	MS
[52]						√			√					√		√		MS

[1] Abbreviation: PS: prior shift; CS: covariate shift; FCS: feature concept shift; LCS: label concept shift; QS: quantity shift; FSH: feature space heterogeneity; LSH: label space heterogeneity; FLSH: feature and label space heterogeneity; HFL: horizontal FL; VFL: vertical federated learning; CFL: centralized FL; DFL: decentralized FL; IA: instance augmentation; IS: instance selection; FA: feature augmentation; FM: feature mapping; FS: feature selection; FC: feature clustering; FAI: feature alignment; CR: consistency regularization; DCR: domain-dependent consistency regularization; PS: parameter sharing; PR: parameter restriction; PD: parameter decoupling; PP: parameter pruning; MW: model weighting; MS: model selection; MC: model clustering; MI: model interpolation; KD: knowledge distillation.

problem, the study [58] suggests using a benchmark model that is trained on a targeted small dataset before FL starts to evaluate the relevance of each participant's data, where only highly relevant data is used for local training. However, this method does not consider the influence of other participants on the benchmark model during FL training. Accordingly, the study [197] finds that calculating the sample loss during FL training can reflect the sample's homogeneity with the global data distribution. Among these, local samples with stronger homogeneity are more conducive to improving the global model's utility. As a result, this study proposes an FL framework named FedBalancer, which employs a selection strategy based on sample loss to filter local samples, aiming to build a local training set that is better aligned with the global sample distribution. However, FedBalancer increases

computational costs because it requires calculating the loss value for every individual local sample. The study [30] introduces a less computationally expensive method for selecting samples. This method uses the gradient upper bound norms of samples to assess their importance to global model performance. It calculates gradients from the loss of the last layer's pre-activation output, rather than calculating the gradient from the overall model parameters. This usually requires just one forward pass to accurately estimate a sample's importance. Specifically, the proposed algorithm includes the following two steps:

1. Participant selection: using the private set intersection (PSI) protocol, each participant is informed about the target categories relevant to the target task. Participants

Table 5   FTL frameworks (continued)

Reference	DHFTL System	DHFTL Incremental	MAFTL	SSFTL	USFTL	PS	CS	FCS	LCS	QS	FSH	LSH	FLSH	HFL	VFL	CFL	DFL	Strategy
[53,55,95,97,137], [100,101,147]	√					√								√		√		MS
[96]						√	√							√		√		MS
[54,99]	√					√			√					√		√		MS
[56]	√		√			√								√		√		MS
[205]	√													√		√		MS
[108,162,206]						√								√		√		MS
[94]						√								√		√		MS,MW
[98]	√					√			√					√		√		MS,MW
[111]	√					√								√		√		MS,MC
[42,45,46,59,64,127]						√								√		√		MC
[43]				√			√							√		√		MC
[169]								√						√		√		MC
[160,168]							√							√		√		MC
[170]										√				√		√		MC
[145]	√					√								√		√		MC
[159]							√			√				√		√		MC
[107]						√								√			√	MC
[171]										√					√			MC
[207]									√							√		MC
[109,161]						√				√				√		√		MC
[44,104]						√								√		√		MC,MI
[105]						√								√		√		MC,KD
[112]						√	√							√		√		MI
[31]						√				√				√		√		MI
[116]					√	√								√		√		MI
[163]			√			√		√						√		√		KD
[32,119,138–140], [33,57,141]						√								√		√		KD
[113,115,144,166]			√			√								√		√		KD
[164]			√			√								√			√	KD
[124]	√		√			√								√		√		KD
[114]	√			√		√				√				√		√		KD
[118,142,143]						√								√		√		KD
[165]						√								√		√		KD
[146]					√	√								√		√		KD

[1] Abbreviation: PS: prior shift; CS: covariate shift; FCS: feature concept shift; LCS: label concept shift; QS: quantity shift; FSH: feature space heterogeneity; LSH: label space heterogeneity; FLSH: feature and label space heterogeneity; HFL: horizontal FL; VFL: vertical federated learning; CFL: centralized FL; DFL: decentralized FL; IA: instance augmentation; IS: instance selection; FA: feature augmentation; FM: feature mapping; FS: feature selection; FC: feature clustering; FAI: feature alignment; CR: consistency regularization; DCR: domain-dependent consistency regularization; PS: parameter sharing; PR: parameter restriction; PD: parameter decoupling; PP: parameter pruning; MW: model weighting; MS: model selection; MC: model clustering; MI: model interpolation; KD: knowledge distillation.

with low relevance to the target task are filtered out. Reversely, if their total number of samples, which match the target categories, reach a certain threshold d, these qualified participants can participate in global aggregation. This prevents participants with large category distribution bias from interfering with the global model's convergence.

2. Sample selection: during tth round, each participant u of the selected participants $\bar{u}$ measures the importance $\lambda(x_{u,i},t)$ of samples $\{x_{u,i}\}_{i=1}^n$ related to the target task's categories, where $\lambda(x_{u,i},t)$ is defined as:

$$\lambda(x_{u,i},t) = \sqrt{|\sum_{u,l} \beta_{u,i}^{t,l} \nabla_{\alpha_{u,i}^{t,l}} f(x_{u,i};\theta_t)|^2},$$

where $\alpha_{u,i}^{t,l}, \beta_{u,i}^{t,l}$ are the input and output of the last layer (lth) of sample $x_{u,i}$ in the tth iteration, respectively. $\sum_{t,l} \beta^l = \text{diag}(\delta'^l(\beta_1),..., \quad \delta'^l(\beta_{r_l}))$. $|\delta'(\beta)| \neq \lambda$ and

$f(x;\theta) := \sum_{u=1}^k \frac{n_u}{n} f_u(\theta)$. β is the output matrix, δ is a trade-off parameter. The importance of a sample is indicated by the value of $\lambda(x_{u,i},t)$: higher values mean higher importance.

Additionally, this selection strategy assumes that there are mislabeled samples locally, and these are often significantly more important than correctly labeled samples [30]. Therefore, by filtering out outlier samples where $\lambda(x_{u,i},t)$ is significantly higher than most other samples, the above algorithm can effectively measure the true distribution of local categories, selecting samples closer to the global distribution for local model training.

● Feature clustering

Feature clustering seeks to find a more abstract representation of original features to group similar data distributions together [7]. In the past, most clustering methods in FTL used model-

related information for clustering [42–46,59,64,127,169,221]. This model-related information includes things like model parameters [46,64,127], gradient information [42,59], training loss [44,169,221], and other external info [43,45]. Except for these, clustering methods based on data-related information also can be applied to mitigate the prior shift issue. For example, Astraea [47] traverses all unassigned participant data distributions by a greedy strategy, looking for a group of participants that can make the overall data distribution of each cluster as close to a uniform distribution as possible.

● Feature selection

For example, Fed-FiS [175] generates local feature subsets on each participant by estimating the mutual information between features and between features and categories. Then, the server ranks each feature and uses classification tasks to obtain a global subset of features. Similarly, research [37] utilizes a federated feature selection algorithm based on MI in the operation of autonomous vehicles (AV). This algorithm completes global iterative feature selection by locally executing an aggregation function based on Bayes' theory, which greatly reduces the computational cost. Moreover, Feature selection is a common idea to extract important features, which can obtain similar performance across different domains, and these important features can serve as a connection for knowledge transfer [7]. In HOFTL, the local dataset of different participants may have similarities in feature space, and high dimensional features can delay the training time, leading to more energy consumption [48]. In this case, removing irrelevant features and selecting useful overlapping features is crucial to address the distribution shift problem in FTL. Current FL researches [37,48,49,175] have proposed a variety of solutions to the above problems, which mainly include three steps:

1. Local filtering: filter the optimal subset of local features of each participant.
2. Global filtering: aggregate local optimal feature subsets to obtain global feature set.
3. Sharing: feed back the global feature set to the participants, allowing participants to focus on the features most relevant to the global representation.

For example, FPSO-FS [49] is a federated feature selection algorithm based on particle swarm optimization (PSO), which proposes two global filtering strategies to determine the global optimal feature subset:

1. Mean assembly strategy: for the private optimal feature subset of the i participant, its average classification accuracy is obtained by the classification accuracy from all participants, i.e., $\text{acc}_{ij}, j = 1, \ldots, k$. Then, an optimal subset with the highest average classification accuracy is selected as the overall optimal feature subset, as follows: $\mathcal{X}^* = \max\{\mathcal{X}_i | \frac{1}{k} \sum_{j=1}^{k} \text{acc}_{ij}(\mathcal{X}_i, \text{Dat}_j), i = 1, \ldots, k\}$, where $\mathcal{X}_i$ is the private optimal feature subset from the ith B participant, Dat_j is the sample data held by the jth B participant, and $acc_{ij}(\mathcal{X}_i, \text{Dat}_j)$ is the classification accuracy of $\mathcal{X}_i$ evaluated by Dat_j.

2. Maximum and minimum assembly strategy: the first step is to identify the minimum classification accuracy for participant i's optimal feature subset, using the classification accuracy $\text{acc}_{ij}, j = 1, \ldots, k$ obtained by all participants. Following this, the subset with the highest minimum classification accuracy among all private optimal subsets is selected as the overall optimal feature subset: $\mathcal{X}^* = \max\{\mathcal{X}_i | \min_{j=1,\ldots,k}(\text{acc}_{ij}(\mathcal{X}, \text{Dat}_j)), i = 1, \ldots, k\}$.

● Consistency regularization

The FTL atrategies also can be explained from a model perspective. Figure 6 shows the corresponding strategies. Among them, consistency regularization [7] refers to the addition of regularization terms to the objective function of local (or global) model optimization, which aims to improve the model robustness of participants, facilitating the transfer of knowledge from the source model to the target model during the training process. In traditional transfer learning, domain adaptation machine [222,223] and consensus regularization framework [224,225] are widely used for knowledge transfer in multi-source domains [7], which are applicable to FL scenarios with two or more participants. The objective function is represented as:

$$\min_{f^T} \mathcal{L}^{T,L}(f^T) + \delta_1 \Omega^D(f^T) + \delta_2 \Omega(f^T),$$

where the first term, as a loss function, is used to minimize the classification error of labeled target domain instances, the second term represents different regularizers, and the third term is used to control the complexity of the final decision function f^T. In addition, according to the research [7], domain-dependent consistency regularization can be expressed as:

$$\min_{f^T} \sum_{j=1}^{n^{T,L}} (f^T(x_j^{T,L}) - y_j^{T,L})^2 + \delta_2 \Omega(f^T)$$
$$+ \delta_1 \sum_{u=1}^{k^S} \lambda_u \sum_{i=1}^{n^{T,U}} (f^T(x_i^{T,U}) - f_u^S(x_i^{T,U}))^2,$$

where λ_u represents the weighting parameter that is determined by the relevance between the target domain and the uth source domain. For example, pFedMe [72] utilizes Moreau envelopes for regularizing participants' loss functions. This approach effectively separates the optimization of individualized models from the learning process of the overarching global model within a structured bi-level framework tailored for FTL. MOON [128] proposes a contrastive learning-based federated optimization algorithm that uses the distribution difference in intermediate outputs between global and local models. Each participant's local optimization goal, beyond the cross-entropy loss term $\mathcal{L}_u^{T,L}(f^T)$, aims to minimize the distance between the local and global model representations (reducing weight divergence) and maximize the distance between the local model and its previous version (accelerating convergence).

● Parameter sharing

The parameters of a model essentially reflect the knowledge that the model has learned. Therefore, in FL, participants can also transfer knowledge at the parameter level [7] by parameter sharing, which avoids the privacy risks brought by direct transmission of local data [1]. For instance, the source and target models share parameters, and the target models use their local data to fine-tune the final layers of the source model, thereby creating a new model [68,69,154]. Since parameter sharing is a common approach in FL and often forms the basis for other methods, this section will focus on explaining the basic method of locally fine-tuning the global model.

Specifically, study [154] finds that fine-tuning global model parameters with local training sets significantly improves prediction accuracy, particularly for local models that differ greatly from global predictions. Unlike FedPer [68], which fine-tunes the global model's top parameters using local data, Per-FedAvg [69] averages all participant model parameters and fine-tunes all global model parameters using the MAML meta-learning method. However, the majority of existing FL frameworks based on parameter sharing mainly focus on improving the global model's performance on each participant using the participants' local data, overlooking the enhancement of a global model's generalization performance from the server's perspective. [117] proposes an FL framework based on a fine-tuning and head model aggregation method, called FedFTHA, which includes FedFT and FedHA. From the participant's perspective, FedFT focuses on improving the performance of the global model to the participant's local dataset by retaining and fine-tuning the local head model. From the server's perspective, FedHA works to reconstruct a global model that exhibits generalized performance, leveraging the participants' head model developed during FedFT. This approach enables both participants and the server engaged in FL to mutually benefit and realize a situation where all parties are advantaged.

● Parameter restriction

The knowledge learned from participants is kept as model parameters and is transferred by the server in CFL. Using the global model directly as the local model usually requires a strong correlation between the global and local data distributions. If there are large differences in data distribution among participants, using the global model directly and optimizing it with local data could lead to a significant decrease in the model's generalization ability. To address this, some studies [71–76,129,202] in FTL restrict the similarity between the source and target models by parameter restriction [7]. For example, FedProx [202] controls the differences between local and global model parameters by a proximal term, which aims to avoid the global model being significantly skewed by too many local updates and further affecting robust convergence. This approach keeps updates close to the initial model, helping to tackle the problem of prior distribution shift and covariate shift issues. However, this proximal term could not align local and global optimal points, and considering the potential loss of important parameter information when the global model is transferred locally. FedCL [130] introduces elastic weight consolidation (EWC) from continual learning [226]. By using a server-side proxy dataset to estimate the importance of global model weights, local updates can be adjusted to prevent significant changes in vital parameters during local adaptive training:

$$\min_{f^T} \mathcal{L}_u^{T,L}(f^T) + \delta \sum_{i,j} \mathcal{R}_{ij} \|\theta_{u,ij}^{T,L} - \theta_{g,ij}^{S,L}\|^2,$$

where $\mathcal{R} = \mathcal{R}_{ij}$ represents the importance matrix of the global model, derived using the server's proxy dataset. FedCL prevents divergence between global and local model weights and ensures better generalization and accuracy. Additionally, FedNova [131] addresses distribution inconsistencies between participants by normalizing and scaling local updates, enhancing model convergence. SCAFFOLD [77] focuses on reducing gradient variance, which first introduces a control variable c_u for the direction of the participant model gradient, and then corrects local model updates based on the difference $(c - c_u)$ between this control variable and a global variable $c(c = \{c_u\}_{(u=1)}^k)$, alleviating shift issue. Additionally, FedCSA [78] adjusts the weights of classifier parameters based on the distribution of each category on the participant. This adjustment enhances performance when dealing with class imbalance.

● Parameter decoupling

Research [227] indicates that the classifier of the model may exhibit significant accuracy decreasing when dealing with imbalanced prior probabilities. Therefore, many studies [227–230] have suggested that sending partial local models for aggregation by decomposing models of participants into body (extractor) and head (classifier) parameters can improve accuracy in the target domain, which is called parameter decoupling. The body parameters can capture general data information and be maintained locally, enabling each participant to learn data characteristics for specific tasks, while head parameters learn specific features of the target domain for sharing with the server to improve the effectiveness of knowledge transfer. For instance, FedRep [79], FedBABU [80], FedAlt [155], FedPer [68] and SPATL [132] perform global aggregation by sharing a homogeneous feature extractor. LG-FedAvg [63], CHFL [133], and FedClassAvg [81] share a homogeneous classifier. Different from these, Fed-ROD [129] shows great effectiveness by splitting the model head into general and personalized layers. Two predictors are trained using a shared body model to handle the competing objectives of generic and personalized federated learning. One predictor employs empirical risk minimization (ERM) for personalization, while the other uses balanced risk minimization (BRM) for general learning. Moreover, FedU [82] proposes a local sharing protocol based on a Siamese network. By aggregating only the online models from the source Siamese network to update the target model, it effectively enables knowledge transfer between participants. However, the study [80] finds that existing parameter decoupling methods by updating the entire model during the training process, lead to a decrease in personalization

performance. Therefore, it proposed an FL framework, called FedBABU. It updates only the body part of the model parameters during the federated training process, and the head is fine-tuned for personalization during the evaluation process. To more accurately determine the degree of impact each layer of the model has on the target domain, the study [83] introduces a layered sharpness-aware Minimization (LWSAM) algorithm, which addresses the problem of poor participant performance due to the biased generic information shared by all participants. This method first calculates the distance between the global and local models at each layer, determining how much each layer is affected by the target domain. It accurately divides the model into head and body parts. Then it uses the sharpness-aware minimization (SAM) algorithm as a local optimizer. By adding more disturbances to the model body, the method adjusts the influence of the target distribution on the model from a global perspective.

● Parameter pruning
Due to varying data distributions among participants, directly applying an aggregated global model to a target domain often does not give optimal results. One popular solution [132] is parameter pruning, which selects a subset of model parameters from the source domain to apply to the target domain. For example, FedMask [85] uses a method called 'model binary masks' to selectively activate certain model parameters for training. This can happen after just one step of communication, without needing to fine-tune the model on local data, which reduces redundancy in communication and computation. In study [132], each participant uses a pre-trained reinforcement learning agent to choose parameters for combining data in a federated manner. They then use a global encoder and a local predictor to transfer knowledge from the combined model to individual models. Another study [86] proposes a federated search method, which uses a lightweight search controller to find an accurate local sub-network for each participant. This method is good at extracting useful information and lowers the energy used for analysis and training. HeteroFL [87] proposes splitting the global model along its width while maintaining the full depth of the participants' DNN models, which aims to find more suitable local models. However, this approach could construct very thin and deep subnetworks, leading to a significant loss of basic features. To overcome this issue, the study [134] introduces a federated learning framework, named ScaleFL, which uses early exits to adaptively reduce DNN models' width and depth, finding models best suited for training with limited local resources.

● Model weighting
The knowledge transfer between participants can be accomplished by sharing local model-related information, such as model parameters, which involves aggregating them before local training, called model aggregation. However, different participants may have distinct optimal goals, simply averaging their model information with the same weight could result in the combined results not being the best solution [1]. The global model in the server may also be overly influenced by a single participant's model, causing 'model drift'

[131,231]. Thus, model weighting is applied to aggregate models according to their contributions. This prevents model performance degradation caused by directly averaging information from different actors into a domain. For example, the study [88] finds local data with higher prediction errors has more contributions to improve the overall model performance, and then introduces an FL framework, called FedCav. FedCav measures the quality of local data using their prediction errors to decide the weights in the model aggregation process. Considering that the server doesn't know the local data distribution, FedFusion [203] uses a global representation of multiple virtual components with different parameters and weights to portray the data distribution of different participants. The server uses a variational autoencoder (VAE) to learn the best parameters and weights of the distribution components based on limited statistical information taken from the original model parameters. Additionally, the study [135] treats the blending of multiple models in FL as a graph-matching task, and then proposes an algorithm, called GAMF. It views channels and weights as nodes and edges of a graph, respectively. Then it uses a new hierarchical algorithm to increase the similarity of weights between channels, and proposes a cycle-consistent multi-graph matching method to merge various local source models in FL, enhancing the global model's generalization. Experiments show that GAMF can be used as a plug-in to boost the performance of existing FL systems.

● Model selection
In reality, participants' local data may significantly differ from the optimal global distribution, and each participant's data characteristics can not be directly controlled, thus it is important to select participants related to the data or specific target labels for training, called model selection.

For example, the study [91] proposes a DFL algorithm based on directed acyclic graphs (DAG). In the DAG, each participant selects the model updates of other participants based on their data similarity, which has been demonstrated effectively in both prior and covariate shift scenarios. Astraea [47], a scheduler-based multi-participant rescheduling framework, re-schedules multiple participants through a scheduler, which follows that the data distribution of multiple participants is most similar to a uniform distribution. Dubhe [148] is an FL algorithm based on repeated model selection, which allows the server to repeatedly select local models to participate in aggregation, and then send the encrypted distribution of the selected participants to the server to check the similarity between the global data distribution after aggregation and the consistency distribution. This process continuously adjusts the aggregation strategy and improves classification accuracy. Another study [60] proposes a Shapley value-based federated averaging algorithm. It calculates the Shapley value of each participant to assess its relevance to the server's learning objective, estimating the participant's contribution in the next communication round of FL. This allows the server to select local models with higher contributions for training in each round of aggregation. In addition, some studies [51,92] require each participant to

collect models from all other participants and use an additional local validation set to evaluate the similarity between participants. In contrast, the study [93] utilizes mathematical analysis methods instead of using empirical search from validation data sets to characterize the similarity between participants. Apart from data distribution, another study [52] considers differences in local data volumes between participants. Note that uniform sampling could overlook participants with more data, reducing their contributions to the global model training and impacting the model's performance. To address this, the researchers propose FedSampling [52], a framework that uses a data uniform sampling strategy. When the participants' data distributions are highly imbalanced, participants randomly select others based on the ratio of the server's desired sample volume to the total available participant sample volume, further improving FL model performance.

● Model clustering

Some studies [44,103] suggest that grouping similar participants for FL can address the model drift issue caused by data distributions' heterogeneity among participants, called model clustering. They determine participant similarity based on factors like model parameters [44,64,104,168,170,232], gradients [42,59], training loss [44,169], or other external information [43,45].

For example, FedCluster [42] uses cosine similarity of the model gradient to split participants into multiple clusters, maximizing similarity within clusters while minimizing it between clusters. To further enhance model adaptability in the target domain by utilizing the sub-model clustering method, the study [105] designs a scale-based aggregation strategy, which scales parameters according to the pruning rate of the sub-models and aggregates overlapping parameters. It further introduces a server-assisted model adjustment mechanism to promote beneficial collaboration between device source models and suppress detrimental collaboration. This mechanism dynamically adjusts the sub-model structure of server devices based on a global view of device data distribution similarity. In addition, studies [43,45] use exogenous information like the types of local devices participants use or patient drug features (drugs given within the initial 48 hours of ICU admission, comprising 1399 binary drug features) for clustering. However, these methods tend to overlook the 'cluster skew' issue caused by grouping, leading to the global model overfitting to a specific cluster's data distribution. "Cluster skew" refers not only to an imbalance in the category distribution among groups after clustering but also to an imbalance in the number of participants in each cluster. To address this issue, study [109] suggests a new FL aggregation method with deep reinforcement learning, called FedDRL, which can tap into the self-learning capability of the reinforcement learning agent, rather than setting explicit rules. Specifically, FedDRL utilizes a unique two-stage training process designed to augment the training data and reduce the training time of the deep reinforcement learning model. Moreover, the study [107] proposes a DFL framework based on hierarchical aggregation, named Spread. In this framework,

the server acts as the FL coordinator, and edge devices are grouped into different clusters. Selected edge devices, as cluster leaders, responsible for model aggregation tasks. Spread monitors training quality and manages model aggregation congestion by adjusting both intra-cluster and inter-cluster aggregations.

● Model interpolation

Different from model weighting methods that combine local models with varying weights, model interpolation aims to blend global and local model parameters proportionally to increase local prediction accuracy [104,110,129]. For example, research [73] prevents the local model and global model from diverging excessively by setting an interpolation coefficient ρ artificially. When ρ is set to 0, the local model only performs local model learning; as ρ increases, the local model gradually becomes similar to the global model, realizing mixed model learning; when ρ is very large, all local models are forced to be similar, maximizing the transfer of knowledge from the global to the local model. Research [112] interpolates a global model trained globally with a local k-nearest neighbors (KNN) model based on the shared representation, which is provided by the global model. The experiments show that it is also suitable for covariate issues except for prior shift issues. However, research [44] demonstrates that these methods, which rely on the separated training of the global model and local model to find the optimal interpolation coefficient ρ^*, may not always be the best. Therefore, it proposes a combined optimization strategy that improves both local and global models at the same time. Similarly, research [31] proposes a model interpolation method based on elastic aggregation. They measure the sensitivity of each parameter by calculating the change in the overall prediction function output when each parameter changes, which reduces the update magnitude for more sensitive parameters, preventing the global model from excessively interfering with the local data distribution.

3.1.2 Covariate shift

● Feature augmentation

Some studies [125,149] have proposed solving covariate shift issues from the perspective of feature augmentation. For example, the study [149] proposes an FL paradigm based on a feature representation generator, called FRAug. It optimizes a common feature representation generator to help each participant generate synthetic feature representations locally, which are converted into participant-specific features by a locally optimized RTNet, which aims to make global and local feature distributions as similar as possible, increasing the training space for each participant. In addition, similar to the study [62] in data augmentation, the study [125] further proposes FedFA to augment features from a statistical perspective. The statistics of augmented data should match as closely as possible with the statistics of the original training data. FedFA follows two preconditions:

1. The data distribution of each participant can be characterized as the statistics of the latent feature distribution, i.e., mean and variance [125].

2. The statistics of latent features can capture basic domain perceptual characteristics [233–235].

Therefore, when dealing with data shift in FL, discrepancies in feature statistics across local participants are inconsistent, and they display uncertain changes compared to the actual distribution's statistics. Based on this, FedFA leverages a statistical probability augmentation algorithm based on a normal distribution to enhance the local feature statistics for each participant following as:

$$\bar{x} = \bar{\sigma} \frac{x_u - \mu_u}{\sigma_u} + \bar{\mu}_u,$$

where the mean μ_u and variance σ_u are the original statistics of the latent features, and $\bar{\mu}_u \sim \mathcal{N}(\mu_u, \bar{\Sigma}_{\mu_u}^2)$, $\bar{\sigma} \sim \mathcal{N}(\sigma, \bar{\Sigma}_{\sigma_u}^2)$. x_u is normalized by $x_u = \frac{(x_u - \mu_u)}{\sigma_u}$, then expanded using the new statistical values $(\bar{\mu}_u, \bar{\sigma}_u)$. The variance $\bar{\sigma}_u$ dictates how much the latent features are augmented. The magnitude of this variance represents how much the latent feature distribution deviates statistically from the target distribution. By adjusting the variance $\bar{\sigma}_u$ appropriately, the skewness issue can be resolved either at the level of individual participants or across all participants. Moreover, it can be integrated as a plugin into any layer of any network to enhance features or solve any feature statistical bias, such as skewness in test time distribution. FedFA also exhibits excellent accuracy performance when addressing prior shift or quantity shift problems of homogeneous FTL.

● Feature clustering
PFA [65] first clusters participants with similar data distribution by computing Euclidean Distance of local representations, and then orchestrates an FL procedure on a group basis to effectively achieve adaptation based on their clustering results. Additionally, the study [167] introduces a new FL framework, called FPL, which is based on prototype clustering. FPL uses the mean features of local data as prototypes, and clusters these prototypes at the server using a comparison learning method. This process brings similar prototypes closer and pushes different ones further apart. Moreover, to increase the stability of model training, FPL uses consistency regularization to minimize the distance between the representative prototypes and their unbiased counterparts. Compared to transferring model parameters in the FPL, the size of the prototypes is much smaller than that of the model parameters, which significantly reduces communication costs.

● Consistency regularization
From the model's perspective, directly adding model-level regularizers to the local objective function of the participants or server is a natural idea [7]. In this way, the knowledge maintained in the model(s) of the participants (source model) can be transferred to the model of another participant (target model) during the training process. For fully supervised learning, each participant first uses their local labeled data to obtain the classification loss term. For example, FedBN [151] keeps participant-specific batch normalization layers to normalize local data distribution. Its classification loss term can be written as:

$$\mathcal{L}^{T,L}(f^T) = \sum_{i=1}^{\bar{k}} \sum_{j=1}^{x_i} (f^T(x_j^i), y_j^i)^2.$$

Except for the classification loss term, a consistency loss term can be introduced based on a cluster-aware mechanism, which uses the differences in both intermediate outputs and predictions between the global and local models to guide local model optimization [153]. It further groups the participants into clusters based on the feature clustering method by harnessing the similarity among lower-level features of each participant's model. Each cluster has its own global feature vector and average prediction value. By minimizing the L2 norm between each participant and the global feature and prediction value within its cluster, the method improves the robustness of local models under the covariate shift issue of homogeneous FTL. Additionally, PFL [167] introduces a consistency regularization term based on a global unbiased prototype. It suggests that the cluster prototype averaged by the server, as an unbiased prototype, can provide a relatively fair and stable optimization point. Calculating the square difference loss between the local feature vector and the unbiased prototype can address the issue of unstable prototype convergence. The regularizer in PFL can be expressed as:

$$\mathcal{L}_{\text{regularizer}} = \sum_{j=1}^{v} (x_{i,j} - \mathcal{U}_j^k)^2,$$

where i and j index samples in dataset of participant u and the dimensions of feature output, respectively. v is the number of dimensions. $\mathcal{U}$ is the unbiased prototype.

● Parameter decoupling
Parameter decoupling is not only suitable for prior shift or quantity shift problems [87], but also for solving covariate shift problems in homogeneous FTL. For example, the study [156] proposes a more flexible way of decoupling, designing an FL algorithm based on structured pruning, called Hermes. In this method, participants determine the sub-networks to participate in server aggregation through model pruning. To prevent information loss that could result from directly averaging local models, Hermes only averages overlapping sub-network parameters on the server, keeping the parameters of the remaining non-overlapping parts unchanged. The aggregated parts of the sub-networks are then sent back to the local devices for network updates, thereby improving the model's performance on local tasks.

● Model weighting
FedUReID [157] enhances the adaptability of the aggregated global model to each participant's local model by applying an exponential moving average (EMA) to update the global model for each participant, where the weight of the EMA represents the similarity between the global and local models. Additionally, FedDG [158] takes advantage of the domain flatness constraint, which serves as a substitute for the complex domain divergence constraint, to approximate the optimal aggregate weights. Moreover, FedDG uses a momentum mechanism to dynamically assign a weight to each isolated domain by tracking the domain generalization gap,

improving its generalization capability. Past studies often simplify the blending of source models into a straightforward allocation problem, ignoring complex interactions between channels. Meanwhile, since model weights are shuffled during training, before merging, channels of each layer need to be aligned to maximize similarities in weights between multiple source models. This presents a quadratic assignment property problem, which is NP-hard problem. To tackle this problem, GAMF [135] propose to treat the channels and weights as nodes and edges of a graph to obtain the weights information. These weighting methods typically rely on model parameters or gradient differences to measure each participant's contribution to the target prediction. However, the transmission of this information involves potential privacy leakage risks. Thus, the study [204] views the local model of each participant as black-box model, in which all data is stored locally and only the source model's input and output interfaces are accessible. Each participant's soft outputs are given a weight based on their inter-class variance. These weighted outputs are then used to create target pseudo-labels. Therefore, it proposes a federated adaptive learning framework called Co-MDA, called CO-MDA. CO-MDA changes the label noise learning section into a semi-supervised learning approach and proposes a Co2-Learning strategy. This strategy involves training two networks at the same time that filter each other's errors through epoch-level co-teaching [236], and gradually co-guess the pseudo-labels with the outputs of both target networks to further reduce the impact of label noise.

● Model clustering
FedDL [170], FedAMP [104], and HYPCLUSTER [44] use model-related information (such as model parameters, convolution layer channels, LSTM hidden states, and neurons in fully connected layers) to construct a shared global model based on model clustering method. However, these methods require several communication rounds to separate all inconsistent participants, potentially affecting computational and communication efficiency. Therefore, the study [168] proposes a method FedMA to achieve hierarchical clustering of participants with a single round of communication. This method uses the difference between the initial global model parameters and local model parameters to generate multiple sub-clusters. Then, by calculating the pairwise distances between participants within all sub-clusters, similar sub-clusters are iteratively merged until only a single cluster remains, containing all samples. Furthermore, similar to study [127] in addressing prior shift issue, study [160] treats the multi-center participant clustering issue as an optimization problem, which can be effectively resolved using the expectation-maximization (EM) algorithm. In addition, different from traditional federated clustering methods, which associate each participant's data distribution with only one cluster distribution (known as hard clustering), the study [159] introduces a soft-clustering-based FL paradigm, called FedSoft. FedSoft allows each local dataset to follow a mixture of multiple cluster distributions, improving the training of high-quality local and cluster models.

● Model selection
Model selection, a classic transfer learning method, has seen widespread use in FTL, either on its own or in combination with other methods, and it is equally effective in addressing covariate shift issues of FTL. For example, CMFL [96] compares the local update of each participant with the global update during each iteration of learning by calculating the proportion of parameters in the local update that have opposite signs to those in the global update, which aims to assess the degree of alignment between the two sets of gradients. A higher proportion indicates a greater deviation from the direction of joint convergence, rendering the local update less relevant. CMFL thus selectively excludes such divergent local updates from being uploaded, effectively minimizing communication costs in FL while ensuring convergence can still be significantly achieved.

3.1.3   Feature concept shift & label concept shift
To mitigate feature concept shift challenges in FTL, study [169] utilizes an iterative federated hierarchical clustering algorithm, called IFCA. Different from traditional methods, IFCA does not require centralized clustering algorithms. The server only plays a role in average model parameters, which substantially decreases the server's computational load. However, IFCA needs to run a federated stochastic gradient descent (SGD) algorithm in each round until it converges. This process could increase computational and communication efficiency in large-scale FL systems.

Regarding the label concept shift issue, current studies address it from the perspectives of feature alignment [150] or model clustering methods [89,207]. Feddg [150] uses the amplitude spectrum in the frequency domain as data distribution information and exchanges it among participants. The goal is that each participant can fully utilize multi-source data distribution information to learn parameters with higher generalization, which proves equally effective under covariate shift. In addition, considering that Bayesian optimization is a powerful surrogate-assisted algorithm for solving label concept shift issues in FL and black-box optimization problems where the local model-related information is not visible to other participants for privacy. Some researchers have turned their attention to federated Bayesian optimization [108,162,206,207]. However, these methods either have all participants work together on the same task, or make only one participant learn from others to tackle a specific task. However, in real life, the tasks of participants are often related to each other. the study [207] introduces an efficient federated multi-task Bayesian optimization framework, called FMTBO, which dynamically aggregates multi-task models based on a dissimilarity matrix derived from predictive rankings. Additionally, FMTBO designs a federated ensemble acquisition function that effectively searches for the best solution by using predictions from both global and local hyperparameters, enhancing the generalization of the global model. Besides, pFEDVEM [89] introduces an FL framework based on Bayesian models and latent variables, and combines the model weighting strategy to mitigate label concept shift issues. In this setup, a hidden shared model identifies common

patterns among local models, meanwhile, local models adapt to their specific environments using the information from the shared model, which determines the confidence levels of each participant. The confidence levels are then used to set the weights when combining local models. The extensive experiments have demonstrated that pFEDVEM robustly addresses three types of distribution shift issues including prior shift, covariate shift, and label concept shift, and obtains significant accuracy improvement compared to the baselines.

3.1.4   Quantity shift

FEDMIX [61] has proven that the instance enhancement method is equally effective for quantity shift problems. Moreover, studies [161,171] view FL as a hedonic game, where each participant produces some cost (error) when joining in the FL process. There's a Nash equilibrium between minimizing individual errors and overall errors. For example, a school may aim to minimize its local error, while a region or city may aim to minimize the overall error. Study [171] proposes to find a relatively stable participant partition by accurately estimating the expected error of each participant, which may overlook the need to minimize the overall error. Additionally, the numbers of samples for participants in [171] are only assumed to be small or large, Different from it, study [161] not only more focuses on overall social well-being, but also is suitable for any number of participants with any various numbers of samples.

3.2   Heterogeneous federated transfer learning

This section will discuss existing works addressing the challenges of heterogeneous FTL, dynamic heterogeneous FTL, and model adaptive FTL from data-based and model-based perspectives as shown in Fig. 6. However, it is worth noting that there is very little research on scenarios with heterogeneous label space and heterogeneous feature and label space, so we will not elaborate on it here.

3.2.1   Feature space heterogeneity

Heterogeneous feature spaces often occur in VFL, thus, we mainly focus on the VFL scenario for feature space heterogeneous FTL. To address this issue, researchers can utilize methods such as feature alignment [172,237,238] or feature concatenation [176] to construct new feature datasets for model training. Among them, feature alignment in VFL can be completed by constructing a novel feature subspace [237], or filling in missing or incomplete features of each participant's feature spaces [172]. These approaches enable knowledge transfer under a homogeneous feature space. Specifically, the study [172] assumes an inconsistency in the feature spaces between two participants, the active participant A and the passive participant B. Both of them map their features $\mathcal{X}_{com}^{B}$ using their respective mapping functions $\theta^{(A,B)}$ and $\theta^{B,A}$ to the same feature space, resulting in new feature representations $\mathcal{X}_{com}^{B} \cdot \theta^{(A,B)}$ and $\mathcal{X}_{com}^{B} \cdot \theta^{(B,A)}$. They optimize the mapping functions $\theta^{(A,B)}$ and $\theta^{(B,A)}$ by minimizing the similarity between the private features $\mathcal{X}_{pri}^{B}$ and $\mathcal{X}_{com}^{B} \cdot \theta^{(A,B)}$, as well as $\mathcal{X}_{pri}^{A}$ and $\mathcal{X}_{com}^{B} \cdot \theta^{(B,A)}$. Finally, participants A and B, through secure bilateral computation, obtain the complete

features $\mathcal{X}_{b}^{A}$ and $\mathcal{X}_{a}^{B}$ respectively. Based on this, the federated aggregation can be implemented under the aligned feature space. However, these methods rely on the existing feature space of participants, ignoring the relationships among these features. By combining feature clustering methods, the active participant can create new, more valuable feature space for knowledge transfer. For example, study [173] proposes a VFL paradigm based on feature space decomposition clustering, called PrADA. The specific steps include:

1. Feature grouping: participant C applies domain expertise to divide raw features into p groups, each containing tightly related features. Moreover, participant C constructs q interactive feature among pairs of feature groups, resulting in a total of h feature groups (where $h = p + q$).
2. Pretraining stage: this stage involves collaborative efforts between source participant B and participant C, to train a set of feature extractors ($f_{\mathcal{E}} = \{f_{\mathcal{E},i}\}$ for $i = 1$ to h) that are capable of learning features which are both invariant across domains and discriminative for labels.
3. Fine-tuning stage: this process is executed in collaboration between active participant A and participant C, with the goal of training participant A's target label predictor C by utilizing the pre-trained set of feature extractors ($f_{\mathcal{E}} = \{f_{\mathcal{E},i}\}$, where $i = 1$ to h).

In addition, PrADA enhances privacy and security using a secure protocol based on partial homomorphic encryption.

However, not all features of the participants are relevant to the task. Therefore, the novel feature space generated by aggregating the features of all parties needs to filter features that are not relevant to the task through feature selection. However, current feature selection techniques [232,239] in distribution learning, often need numerous training iterations, particularly when dealing with high-dimensional data. Directly applying them in FTL to solve feature space heterogeneous issues produces significant computational and communication overhead, as each training round involves multiple encryptions, decryption operations, and intermediate parameter transfers. For example, study [179] suggests using an embedded method to combine autoencoders with L2 constraints on feature weights for feature selection, and sets a threshold for post-training to determine the selected features for mitigating the problem of model parameter shrinkage [240]. Different from previous VFL research scenarios where there were mostly two participants and binary classification tasks, study [178] proposes an FL feature selection scheme suitable for multi-participant multi-classification. In addition, previous studies that mainly focus on the relationship between features and labels [180], ignoring the relationship between features, to solve this problem, similar to research [37,175] using MI theory into federated feature selection in HFL, study [174] proposes a feature selection VFL framework based on conditional mutual information, called FEAST. FEAST integrates feature information into a single statistical variable for FL transmission, which not only accomplishes key feature selection and further reduces communication costs while

ensuring privacy and security. In addition, study [181] first proposes a theoretically verifiable feature selection method, formalizing the feature selection problem in the VFL environment, and providing a theoretical framework to prove that unimportant features have been removed.

3.3   Dynamic heterogeneous FTL

3.3.1   System heterogeneity

System heterogeneity among participants could lead to the emergence of stragglers in each iteration. To address this problem, instance selection can be leveraged by researchers to mitigate the computational burden of participants when they have heterogeneous local computational resources. However, it could lead to decreased model performance due to the reduced statistical utility of the training dataset. Study [198] obtains the optimal data selection scheme through an optimization function that includes lower and upper limits of resources, as well as arbitrary, non-decreasing cost functions per resource, meanwhile, it treats this problem as a scheduling problem of tasks assignment to resources, seeking to maximize the number of participants in each round of FL updates. In addition, FedBalancer [197] chooses samples for training by measuring their statistical utility, derived from the sample loss list based on the latest model. However, it is inefficient to wait for every participant to finish local training before proceeding with aggregation due to system heterogeneity. Thus, under a constant FL round deadline setting, instance selection could not immediately enhance the time-to-accuracy ratio.

Furthermore, model-based strategies, such as consistency regularization [202], model selection [1], model clustering [55,97,98,145], parameter decoupling [106], parameter pruning [85] can also be applied as effective ways to address the straggler issue in FL. For example, Fedavg [1] directly drops models of these stragglers which can not accomplish local training in time when the other participants have completed the same amount of training. Based on this, FedProx [202] allows varying local training epochs across participants, tailored to each device's system capabilities. Subsequently, it aggregates the non-convergent updates submitted by stragglers, rather than dropping these less responsive participants from this iteration. However, these frameworks may come at the cost of sacrificing accuracy due to the omission of partial information and waiting for all participants to complete a uniform number of training epochs tends to extend the convergence time of FL. FedAT [106] blends synchronous and asynchronous updates by decoupling the model parameters at the layer level, which stratifies local models based on the time each participant needs to complete a round of training. During each training round, FedAT randomly selects some local models in each layer to calculate the loss gradient of local data, completing the synchronous update of models in that specific layer. Each layer, acting as a new training entity, then asynchronously updates the global model. The faster layers have shorter round-response delays, speeding up the convergence of the global model. The slower layers contribute to global training by asynchronously sending model updates to the server, which further improves the predictive performance of the model. Besides, [85,86,134] selectively use local models for transfer knowledge by parameter pruning methods under the limited computational resources. Study [145] introduces a FL framework, called FedHiSyn, which uses a resource-based hierarchical clustering approach. This framework first categorizes all available devices according to their computing capabilities. Given that a ring topology is more suitable for models with uniform resources, after local training, the models are sent to the server. Then, within their respective categories, they exchange local model weight updates based on the ring topology structure to mitigate the lag effect caused by system heterogeneity. Considering that the main challenge of dynamic heterogeneous FTL is the appropriate scheduling of participants, essentially a local model selection issue at each iteration. Some researchers [53,94–96,136,137] assume the central party has 1-lookahead in source model selection strategies, which means that the dynamic input data beforehand is known. However, this can not be applied when dealing with unpredictable time series inputs. Thus, reinforcement learning has been increasingly used to design source model selection strategies [54,94,97,111,132]. Studies [97,98] propose FL paradigms based on the multi-armed bandit (MAB). In situations where the available computing resources of participants are unknown, these approaches calculate the difference between the data distribution of multiple combined participants and a class-balanced data distribution, and then pick local models for aggregation and assign weights based on these differences. In another study, Study [55] proposes a participant scheduling strategy by age of update (AoU) measurement. This strategy considers the age of the received parameters and the current channel quality at the same time, which improves efficiency and allows effective aggregation in federated joint learning.

Except for the above-mentioned reinforcement learning methods in dynamic heterogeneous FTL, TiFL [99], a hierarchical federated learning framework, addresses system heterogeneity issues by grouping participants with similar training performance. Meanwhile, in each round of updates, TiFL adaptively selects local models for training within each group by simultaneously optimizing accuracy and training time. To accelerate model convergence under system heterogeneity, FedSAE [100] suggests choosing participants with larger local training losses to take part in aggregation during each training round. FedSAE also designs a mechanism to predict each participant's maximum tolerable workload, aiming to dynamically adjust their local training rounds. Research [205] found that differences in bit-width among participant devices can impact the performance of the global model. Low-bit-width models are more compatible with hardware but could limit the model's generalization, leading to poor performance of high-bit-width models. To tackle this, the study [205] introduces ProWD, a FL framework that considers bit-width heterogeneity. This framework selects sparse sub-weights compatible with full-precision model weights from the low-bit-width models received by the server. These selected sub-weights then participate in central aggregation along with the full-precision model weights. Another study

[101] proposes a FL framework, called Aergia, which freezes the most computation-heavy parts of the model and trains the unfrozen parts. Moreover, the server chooses a more reasonable offloading solution based on the training speed reported by each participant and the similarity between their datasets. In this way, the training of the frozen parts can be offloaded to participants with ample resources or faster training speeds. PyramidFL [147] is a fine-grained participant selection, which considers not only the distribution and system heterogeneity between the selected and non-selected participants but also within the selected participants themselves. Specifically, the server uses feedback from past training rounds to rank participants based on their importance, participants then use their rank to determine the number of iterations for data efficiency and the parameters to drop for system efficiency. Furthermore, the utility of each participant isn't static but varies across training rounds. If a participant is selected, its data utility will then decrease since these data have been seen by the model. Thus, reducing the likelihood of selection in subsequent training rounds allows participants who were not selected previously to have a higher probability of being chosen, which further improves the model performance under the dynamic heterogeneity and further enhances the fairness of participant selection.

3.3.2   Incremental heterogeneity

Existing FTL strategies mainly focus on model-based techniques, for example, GLFC [102] addresses the continuous emergence of new classes in federated online learning by consistency regularization method. It introduces a class-aware gradient compensation loss to ensure the consistency of the learning pace for new classes with the forgetting pace of old classes. It separately normalizes the gradients for new and old classes and reweights them for the local optimization goal, where the relationships between classes are obtained based on the best old classification model from the previous tasks. As the local data or tasks increase, where the tasks may be a new batch of data, the task $\mathcal{T}(t)_i$ learned by participant u_i in round t could be similar or related to the task $\mathcal{T}(t+1)_j$ learned by participant u_j in round $t+1$. In this situation, the transmission of aggregated global information among participants can facilitate knowledge transfer across participants. Nonetheless, when a new joined task in participant u_j is irrelevant to the tasks of participant u_i, it could affect the optimization direction of the local model, leading to a decrease in accuracy. To mitigate this issue, each participant selectively utilizes only the knowledge of the relevant tasks that have been trained on other participants during each iteration, while ignoring as much as possible the knowledge of irrelevant tasks that may interfere with local learning. Thus, parameter decomposition and model weighting methods have attracted the attention of researchers [110]. For example, FedWeIT [110] solves this problem by decomposing parameters into three different types for training: global parameters that capture the global and generic knowledge across all participants, local base parameters that capture generic knowledge for each participant, and task-adaptive parameters for each specific task per participant. Meanwhile,

FedWeIT applies sparse masks to select parameters relevant to a given task, minimizing interference from irrelevant tasks of other participants and allocating attention to the server's aggregated parameters to selectively filter parameter information. However, all these methods lack a theoretical basis for ensuring convergence. FedL [56] uses dynamic adaptation to measure the extent to which online decision constraints are breached and calculates a maximum limit for this measure, which guarantees that the expected contribution of the chosen source model to the FL model's performance matches its actual contribution.

3.4   Model adaptive FTL

Model adaptive FTL is often caused by model heterogeneity, i.e., the local models of different participants may be inconsistent in architecture, which could cause incompatibility in the feature dimension and representational capacity among participants [115]. It means that the average aggregation approach based on consistent features can not be used directly in FL. To mitigate this issue, data-based strategies are proposed in FTL. For example, the study [200] proposes to apply a feature mapping method to obtain consistent representation space and complete FL. Considering that even with different model structures, they possess some common knowledge for the same input, i.e., the feature extraction layers generate similar feature maps, research [200] uses "model drafts" to align local data distributions of participants. These outputs from specific layers or models are interpreted as blurred images of data and defined as model drafts. By minimizing the similarity difference between the local and global drafts, the data distribution difference between participants can be reduced. Except for feature mapping [7,126] using explicit features, some implicit features can be utilized to align the source and target domains, facilitating knowledge transfer within this aligned space [7], called feature alignment. Implicit features include subspace attributes [201], spectral characteristics [150], prototype graphs [66]. For example, the study [66] uses prototypes to effectively transfer information under the model adaptive FTL by minimizing local and global prototype graphs within the same feature space, thereby capturing the semantic information of class structures. It avoids the possibility of data from different classes (across various participants) merging into a single class, or data from the same class being spread across multiple classes. Similarly, FedHeNN [103] is a FL framework based on instance-level representation. Each participant randomly selects part of local data and obtains instance-level representation to guide local training. By introducing a distance function based on centralized kernel alignment as a proximal term of the local loss function, it aligns local and global model representations, enabling federated learning across heterogeneous models. FedFoA [201] adds a linear calibration layer at the end of each local model to first calibrate the different feature dimensions among participants to the same dimension space. Participants use QR decomposition to obtain feature subspace R and feature correlation matrix Q. The central server minimizes the reconstruction of local features X and the product of global R

and initial vector Q to get the optimal Q^*, guiding the update of the local sub-feature space.

Inconsistent model architectures also make the simple average aggregation of model parameters ineffective. From the model-based perspective, studies [87,156] implement FedAvg on top of local sub-networks by parameter decoupling. [87] assumes that the model architectures of participants can dynamically change with each iteration, and further proposes to leverage parameter decoupling to obtain at least one fixed sub-network for each type of heterogeneous situation and aggregate them into a single global model. Thus, smaller local models can gain more from global aggregation by performing less global aggregation on a subset of the parameters from larger local models. Similarly, study [84] combines parameter decoupling with model clustering method to group local models based on the similarity of their personalized sub-networks, maximizing the level of knowledge sharing between participants. Besides, research [144] designed a Mapper at the local level to convert feature representations from different semantic spaces to the same feature space, and accomplish the knowledge transfer based on knowledge distillation (KD). They deploy a global generator at the server to extract global data distribution information and distill it into the local model of each participant. Then, local models are viewed as discriminators to reduce the difference between global and local data distributions in heterogeneous feature spaces. Since the feature representations synthesized by the global generator are usually more faithful and homogeneous to the global data distribution, they can achieve faster and better convergence. Additionally, local generators also can be used to enhance hard-to-judge sample data, improving model performance [144]. In real-world scenarios, cross-institutional FL is often more content with the VFL scenario. However, traditional VFL can only benefit from samples shared among multiple parties, which severely limits its application. Therefore, research [115] proposes a VFL framework based on representation distillation, called VFedTrans. This framework collaboratively models common features among multiple parties and extracts federated representations of shared samples, aiming to maximize data utility as much as possible through KD.

Knowledge distillation is also often used for dealing with model adaptive FTL induced by model heterogeneity as shown in Table 5. FedMD [163] uses KD for federated learning in situations where different models are used. Instead of just combining model parameters, FedMD calculates class scores for each participant using a shared dataset. These scores are then sent to a server to calculate an average, which guides the training of local models. This method allows for knowledge sharing while keeping private data and model structures secure, and it works even when different local models are used. Contrary to the assumption that participants' local models are entirely different, research [119] assumes that local models of participants are not fully heterogeneous, and there are cases where some models share the same structure. Therefore, they propose an FL framework based on ensemble distillation, called FedDF. It creates several prototype models, which represent participants with identical model structures.

In each round, FedAvg is performed among participants with the same prototype model to initialize a global model (student model), followed by cross-architecture learning through knowledge distillation. In this process, the parameters of the local model (teacher model) are tested on an unlabeled public dataset to generate predictions for training each student model on the server. Research [113] adopts a federated communication strategy, denoted as FSFL, which is similar to FedMD, innovatively adding a latent embedding adaptive module to alleviate the impact of domain discrepancies between public and private datasets. However, these studies [32,33,57,114,119,139–141,163,164,241] strongly rely on the construction of public datasets, which undoubtedly compromises data privacy and is operationally challenging in practice. The impact of the quality of these prerequisites on the performance of federated learning is also unknown [124]. Therefore, research [124] proposes an FL framework based on zero-shot knowledge distillation, called FedZKT. FedZKT requires no prerequisites for local data, and its distillation tasks are assigned to the server to reduce the local workload. In addition, some studies [29,115,142–144,165] introduce generators to avoid the need for public datasets, enabling the aggregation of local information in a data-free manner. For example, FedFTG [143] uses the log-odds of each local model as a teacher to train a global generator and fine-tunes the global model using pseudo-data generated by the global generator. Due to the additional computational and communication costs imposed by the introduction of a generator, ScaleFL [134] proposes a self-distillation method based on exit predictions. This method treats self-distillation as an integral part of the local training process, requiring no extra overhead. ScaleFL enhances the knowledge flow among local sub-networks by minimizing the Kullback–Leibler divergence (KL divergence) between early exits (students) and final predictions (teachers). Furthermore, the study [166] introduces a FL framework based on a data-free semantic collaborative distillation, called MCKD. This framework transfers soft predictions from local models to a server to learn representations that don't change across different domains. MCKD also introduces a knowledge filter to mitigate the potential amplification of irrelevant or malicious participants' influences on the target domain by traditional averaging aggregation. This knowledge filter generates consensus knowledge for unlabeled data and sets a threshold to drop models where local model predicted classes are inconsistent with consensus classes, further adapting the central model to target data. However, most of these methods construct ensemble knowledge by merely averaging the soft predictions of multiple local models, overlooking that local models have a differential understanding of distillation samples. Research [138] suggests that a model is more likely to make the correct predictions when the samples are included in the domain used for the model's training. Based on this, the study [138] proposes treating each participant's local data as a specific domain and designs a domain-aware federated distillation method named DaFKD. DaFKD can recognize the importance of each model to the distillation samples. For a given distillation sample, if the local model has a significant

relevance factor with it, DaFKD assigns a higher weight to this local model.

3.5 Semi-supervised and unsupervised FTL

The labeled data scarcity is a common scenario in both HFL and VFL scenarios. Data-based methods receive some attention in reality for SSFTL issues, some studies [120,121] utilize these methods to obtain valuable augmented data by strategically optimizing predictions or similarity calculations on these data. For example, SemiFed [120] uses both past local and global models to predict labels for local unlabeled data. When predictions have high confidence, the pseudo-labeled data is used to train as available data. Orchestra [121] uses a feature clustering method to obtain more available data for training. It refines local data clusters through interactions with the server, which helps find effective samples in unlabeled data, further increases sample size, and reduces distribution heterogeneity when labeled data is scarce. Different from them without considering the high-class imbalance of unlabeled data, based on the instance selection method, CBAFed [123] uses the empirical distribution of all training data from the last round of global communication to design category-balanced adaptive thresholds. That is, if the model uses more data for training in one class, the threshold for labeling unlabeled data in this class will increase, and vice versa. It aims to shrink the gap between local and global distributions by selecting a local training set, and further influence category distribution, preventing a decline in global model performance due to prior probability bias.

Model-based strategies similarly show the effectiveness in addressing the SSFTL issue, study [50] introduces a framework based on the consistency regularization method, called FedMatch. This framework splits local model parameters into two parts, one for updating with labeled data, and another for unlabeled data. For the parameters associated with labeled data, the loss function solely includes cross-entropy loss, and the loss function related to unlabeled data follows as:

$$\min_{f^{T,U}} \mathcal{L}^{T,U}(f^{T,U}) + \delta_{L_2}\|\theta_L^* - \theta_U\|_2^2 + \delta_{L_1}\|\theta_U\|_1,$$

where L_1- and L_2- regularization on θ_U aims to make θ_U sparse, while not drifting far from the current optimal parameter θ_L^* trained with labeled data. The first term differs from the studies [120,152], which not only uses the prediction results of unlabeled data and its augmented data to obtain cross-entropy loss but also considers pseudo-labels as real labels to obtain new category loss, making the model perform the same on original data and slightly perturbed data.

Contrary to previous studies that assume the presence of both labeled and unlabeled data locally, research [50] describes a federated semi-supervised learning scenario where some participants have fully labeled data while others have only unlabeled data. To guide unlabeled participants' learning by building the interaction between the learning at labeled and unlabeled participants [120,152], FedIRM [41] applies the intermediate output of the local model, which is trained by the labeled participants, to construct a category relationship matrix. By transmitting the relationship matrix of each labeled participant, it guides the unlabeled participants to learn their local relationship matrix. The formula is as follows:

$$\min_{f^{T,U}} \delta(w)(\mathcal{L}^{T,U}(f^{T,U}) + \mathcal{L}_{\mathrm{IRM}}),$$

$$\mathcal{L}_{\mathrm{IRM}} = \frac{1}{z}\sum_{j=1}^{z}(\mathcal{L}_{\mathrm{KL}}(\mathcal{R}_j^L\|\mathcal{R}_j^U) + \mathcal{L}_{\mathrm{KL}}(\mathcal{R}_j^U\|\mathcal{R}_j^L)),$$

where $\mathcal{R}_j$ denotes the relation vector of class j. Furthermore, researchers [67,128] also apply domain-dependent consistency regularization to solve this issue. For example, [67] uses the KL divergence between local and global model predictions to control the update of the local objective function. Additionally, other model-based strategies, such as parameter decoupling [50], model weighting [90,122,152], also attract some researchers' attention. FedMatch [50] accomplishes the FTL under semi-supervised learning scenarios by training labeled and unlabeled data separately through model parameter decomposition. FedConsist [152] and RscFed [122] improve the performance of models in semi-supervised FTL by changing the weights of participants with labeled and unlabeled data. Since semi-supervised learning often deals with data that only has positive and unlabeled (PU) samples, one participant's negative class could be made up of multiple positive classes from other participants. However, traditional PU learning mostly focuses on binary problems with only one kind of negative sample. Thus, study [90] assumes that the available data has multiple types of positive and negative classes (MPMM-PU), and further proposes to redefine the expected risk of MPMM-PU, which aims to decide each participant's weight and examines the limits of the model's generalization.

In response to unsupervised FTL, a straightforward approach is to combine self-supervised methods with FL, such as [195,242]. However, this challenge is often accompanied by homogeneous or heterogeneous FTL issues. The feature-based strategies have obtained some attention to solve this problem, such as FedCA [199] based on feature selection, FSHFL [48] based on feature mapping, and FedFoA [201] based on feature augmentation. Moreover, studies also propose to alleviate the data drift issues between participants by model-based strategies, such as FedX [146] based on KD, and FedEMA [116] based on model interpolation. FedCA [199] shares features of local data and employs an auxiliary pubic dataset to minimize disparities in the representation space across participants. However, it ignores the inconsistency between the feature representation of local unlabeled data and global feature representation. To solve this problem, FSHFL [48], an FL framework based on an unsupervised federated feature selection approach, proposes the feature cleaning locally and global feature selection. The local feature cleaning utilizes an enhanced version of the one-class support vector machine (OCSVM) algorithm, called FAR-OCSVM, to identify features that lack sufficiently representative global features. The identification relies on local feature clustering, features within each cluster exhibit strong interrelationships, thus the clusters with more features contribute more significantly to the FTL process. Meanwhile,

the server selects global features from the collected local feature sets and then returns these global features to participants, directing them to select local features that are closest to the global representation. Considering that the public dataset has potential privacy leakage risk [199], FedX [146] incorporates local knowledge distillation and global knowledge distillation into the FedAvg [1] without any public data. Local knowledge distillation trains the network using the feature representations of local unlabeled datasets, and global knowledge distillation aims to mitigate data shift, which only relies on global model sharing. Besides, FedEMA [116] utilizes self-supervised learning methods with predictors, including MoCo and BYOL, to update local encoders through the EMA of the global encoder.

4   Application

In this section, we explore and outline the prevalent applications where FTL makes a significant impact.

4.1   Federated cross-domain recommendation

Cross-domain recommendation (CDR) aims to reduce data sparsity by transferring knowledge from a data-rich source domain to a target domain. However, this process presents significant challenges related to data privacy and knowledge transferability [243,244]. Therefore, federated learning is introduced to CDR to improve the performance of the target domain model while providing privacy protection. Currently, classic recommendation algorithms have been widely applied to federated learning, such as federated collaborative filtering [245,246], federated matrix factorization [247–249], and federated graph neural networks [250]. However, these methods neglect the heterogeneity among them including data, resource, or model heterogeneity. Researchers have combined transfer learning techniques to accomplish tasks related to federated cross-domain recommendation. They utilize parameter sharing [251], parameter decoupling [252,253], and model clustering [254] to facilitate the process.

4.2   Federated medical image classification

Medical data involving patient information is sensitive and its use is strictly regulated, limiting the application of current artificial intelligence technologies in the medical field. Federated learning, which trains models on local devices without sharing raw data, could protect patients' data privacy security. However, data provided by different medical institutions, acting as source domains, often have heterogeneity in format, features, or distribution [255]. By integrating FL with transfer learning, we can leverage data from different healthcare institutions for model training, enhancing the model's performance while preserving data privacy. This approach presents a promising direction for the application of artificial intelligence in the medical field. Common methods include federated knowledge distillation [139,256], federated weighting aggregation [204,257–260], federated consistency regularization [261,262], federated model selection [263], federated model interpolation [264], federated model decoupling methods [265], federated model clustering [266], and feature clustering methods [267].

4.3   Federated financial service

Federated financial services include credit risk control [268–272], stock prediction [273–275], financial fraud transaction detection [276,277], etc. Among them, credit risk control is a standard procedure for financial institutions, which estimates whether an individual or entity is able to make future required payments [269]. Stock prediction allows economists, governors, and investors to model the market, manage the resources and enhance stock profits [273]. Fraudulent transaction detection is another difficult problem for individual banks to curb company financial losses and maintain positive customer relationships [276]. However, their predictive models all rely on large amounts of data to achieve training. Since the customers' data or fraudulent transaction data can not be shared among different institutions, the prediction model of each financial institution often suffers from the limited sample issue, which significantly hinders the performance improvement of the model. The emergence of federated learning not only breaks this "data silos" problem but also can combine instance augmentation [271,275], instance selection [270], feature selection [268,269,277], feature alignment [237,238], parameter sharing [272], model weighting [273], model selection [274,276], knowledge distillation [272], and other technical approaches to achieve information exchange between institutions.

4.4   Federated traffic flow prediction

Given that precise and up-to-date traffic flow data is of immense significance for traffic management, predicting traffic flow has emerged as a crucial element of intelligent transportation systems. However, current traffic flow prediction techniques, which rely on centralized machine learning, require the collection of raw data from mobile phones and cameras for model training, causing potential privacy issues. Researchers found that this issue also can be addressed using FTL, such as federated clustering aggregation [278–283], federated weighting aggregation [284–286], federated parameter sharing [285], and federated parameter control methods [283].

5   Conclusion and future work

This survey provides a systematic summary of federated transfer learning, which emerges from the combination of transfer learning with federated learning, offering the corresponding definitions, challenges, and strategies. For convenience for researchers, we compile settings for the most common data heterogeneity scenarios, including both homogeneous and heterogeneous federated transfers, and summarize some significant FTL studies for various challenges.

In the future, the utility performance of FL models in more complex scenarios deserves further exploration, including label concept shift, feature concept shift, label space heterogeneity, and feature & label space heterogeneity. Due to differences or mutations in unknown or hidden relationships between input and output variables between participants, the concept shift becomes an important but less studied direction in FL. Meanwhile, label space heterogeneity and feature &

label space heterogeneity also deserve more in-depth study.

Furthermore, although federated transfer learning uses model selection, weight aggregation and other transfer learning methods to solve the problems of data heterogeneity, dynamic heterogeneity and labeled data scarcity in FL, new privacy concerns may have also emerged about these strategy preferences. After participants crack the weighting strategy or selection mechanism used by the server to aggregate information from all parties, they may use unfair means to make the training of the global model develop in a direction more beneficial to themselves. Thus, possible leakage of strategy preferences should be properly considered when designing FTL strategies in the future.

Finally, the communication costs, communication efficiency, and computational costs caused by these FTL strategies could be paid more attention by FL researchers. Compared with traditional FL methods, FTL strategies, such as instance augmentation, instance selection, feature selection, model selection, and model clustering, etc., increase additional computational costs and introduce more shared information among participants. Based on some traditional methods [287−290] for enhancing computational or communication efficiency, FTL researchers can further optimize federated algorithms. In summary, FTL still has great potential research value in utility, privacy, and communication issues.

Competing interests The authors declare that they have no competing interests or financial conflicts to disclose.

References

1. McMahan B, Moore E, Ramage D, Hampson S, Arcas B A Y. Communication-efficient learning of deep networks from decentralized data. In: Proceedings of the 20th International Conference on Artificial Intelligence and Statistics. 2017, 1273−1282

2. Feng S, Li B, Yu H, Liu Y, Yang Q. Semi-supervised federated heterogeneous transfer learning. Knowledge-Based Systems, 2022, 252: 109384

3. Zhang C, Xie Y, Bai H, Yu B, Li W, Gao Y. A survey on federated learning. Knowledge-Based Systems, 2021, 216: 106775

4. Gao L, Fu H, Li L, Chen Y, Xu M, Xu C Z. FedDC: Federated learning with non-IID data via local drift decoupling and correction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022, 10102−10111

5. Shi Y, Zhang Y, Xiao Y, Niu L. Optimization strategies for client drift in federated learning: a review. Procedia Computer Science, 2022, 214: 1168−1173

6. Liu Y, Kang Y, Zou T, Pu Y, He Y, Ye X, Ouyang Y, Zhang Y Q, Yang Q. Vertical federated learning: concepts, advances and challenges. 2022, arXiv preprint arXiv: 2211.12814

7. Zhuang F, Qi Z, Duan K, Xi D, Zhu Y, Zhu H, Xiong H, He Q. A comprehensive survey on transfer learning. Proceedings of the IEEE, 2021, 109(1): 43−76

8. Yang Q, Liu Y, Chen T, Tong Y. Federated machine learning: concept and applications. ACM Transactions on Intelligent Systems and Technology, 2019, 10(2): 12

9. Rahman K M J, Ahmed F, Akhter N, Hasan M, Amin R, Aziz K E, Islam A K M M, Mukta M S H, Islam A K M N. Challenges, applications and design aspects of federated learning: a survey. IEEE Access, 2021, 9: 124682−124700

10. Liu J, Huang J, Zhou Y, Li X, Ji S, Xiong H, Dou D. From distributed machine learning to federated learning: a survey. Knowledge and Information Systems, 2022, 64(4): 885−917

11. Li L, Fan Y, Lin K Y. A survey on federated learning. In: Proceedings of the16th IEEE International Conference on Control & Automation (ICCA). 2020, 791−796

12. Zhan Y, Zhang J, Hong Z, Wu L, Li P, Guo S. A survey of incentive mechanism design for federated learning. IEEE Transactions on Emerging Topics in Computing, 2022, 10(2): 1035−1044

13. Yin X, Zhu Y, Hu J. A comprehensive survey of privacy-preserving federated learning: A taxonomy, review, and future directions. ACM Computing Surveys, 2022, 54(6): 131

14. Lyu L, Yu H, Yang Q. Threats to federated learning: a survey. 2020, arXiv preprint arXiv: 2003.02133

15. Pfitzner B, Steckhan N, Arnrich B. Federated learning in a medical context: a systematic literature review. ACM Transactions on Internet Technology, 2021, 21(2): 50

16. Nguyen D C, Pham Q V, Pathirana P N, Ding M, Seneviratne A, Lin Z, Dobre O, Hwang W J. Federated learning for smart healthcare: a survey. ACM Computing Surveys, 2023, 55(3): 60

17. Lim W Y B, Luong N C, Hoang D T, Jiao Y, Liang Y C, Yang Q, Niyato D, Miao C. Federated learning in mobile edge networks: a comprehensive survey. IEEE Communications Surveys & Tutorials, 2020, 22(3): 2031−2063

18. Nguyen D C, Ding M, Pathirana P N, Seneviratne A, Li J, Poor H V. . Federated learning for internet of things: a comprehensive survey. . IEEE Communications Surveys & Tutorials, 2021, 23(3): 1622−1658

19. Zhu H, Xu J, Liu S, Jin Y. Federated learning on non-IID data: a survey. Neurocomputing, 2021, 465: 371−390

20. Tan A Z, Yu H, Cui L, Yang Q. Towards personalized federated learning. IEEE Transactions on Neural Networks and Learning Systems, 2023, 34(12): 9587−9603

21. Pan S J, Yang Q. A survey on transfer learning. IEEE Transactions on Knowledge and Data Engineering, 2010, 22(10): 1345−1359

22. Konečný J, McMahan H B, Ramage D, Richtárik P. Federated optimization: distributed machine learning for on-device intelligence. 2016, arXiv preprint arXiv: 1610.02527

23. Long M, Wang J, Sun J, Yu P S. Domain invariant transfer kernel learning. IEEE Transactions on Knowledge and Data Engineering, 2015, 27(6): 1519−1532

24. Long M, Cao Y, Wang J, Jordan M I. Learning transferable features with deep adaptation networks. In: Proceedings of the 32nd International Conference on Machine Learning. 2015, 97−105

25. Bengio Y. Deep learning of representations for unsupervised and transfer learning. In: Proceedings of 2011 International Conference on Unsupervised and Transfer Learning Workshop. 2011, 17−37

26. Raina R, Battle A, Lee H, Packer B, Ng A Y. Self-taught learning: transfer learning from unlabeled data. In: Proceedings of the 24th

International Conference on Machine Learning. 2007, 759−766

27.　Younis R, Fisichella M. FLY-SMOTE: re-balancing the non-IID IoT edge devices data in federated learning system. IEEE Access, 2022, 10: 65092−65102

28.　Wu Q, Chen X, Zhou Z, Zhang J. FedHome: cloud-edge based personalized federated learning for in-home health monitoring. IEEE Transactions on Mobile Computing, 2022, 21(8): 2818−2832

29.　Jeong E, Oh S, Kim H, Park J, Bennis M, Kim S L. Communication-efficient on-device machine learning: Federated distillation and augmentation under non-IID private data. 2018, arXiv preprint arXiv: 1811.11479

30.　Li A, Zhang L, Tan J, Qin Y, Wang J, Li X Y. Sample-level data selection for federated learning. In: Proceedings of the IEEE Conference on Computer Communications. 2021, 1−10

31.　Chen D, Hu J, Tan V J, Wei X, Wu E. Elastic aggregation for federated optimization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2023, 12187−12197

32.　Chen H Y, Chao W L. Fedbe: Making Bayesian model ensemble applicable to federated learning. In: Proceedings of the 9th International Conference on Learning Representations. 2021

33.　Seo H, Park J, Oh S, Bennis M, Kim S L. Federated knowledge distillation. 2020, arXiv preprint arXiv: 2011.02367

34.　Gong B, Shi Y, Sha F, Grauman K. Geodesic flow kernel for unsupervised domain adaptation. In: Proceedings of 2012 IEEE Conference on Computer Vision and Pattern Recognition. 2012, 2066−2073

35.　Peng X, Bai Q, Xia X, Huang Z, Saenko K, Wang B. Moment matching for multi-source domain adaptation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019, 1406−1415

36.　Kuznetsova A, Rom H, Alldrin N, Uijlings J, Krasin I, Pont-Tuset J, Kamali S, Popov S, Malloci M, Kolesnikov A, Duerig T, Ferrari V. The open images dataset V4: Unified image classification, object detection, and visual relationship detection at scale. International Journal of Computer Vision, 2020, 128(7): 1956−1981

37.　Cassara P, Gotta A, Valerio L. Federated feature selection for cyber-physical systems of systems. IEEE Transactions on Vehicular Technology, 2022, 71(9): 9937−9950

38.　Darlow L N, Crowley E J, Antoniou A, Storkey A J. CINIC-10 is not ImageNet or CIFAR-10. 2018, arXiv preprint arXiv: 1810.03505

39.　Liu Z, Luo P, Wang X, Tang X. Deep learning face attributes in the wild. In: Proceedings of the IEEE International Conference on Computer Vision. 2015, 3730−3738

40.　Reddi S J, Charles Z, Zaheer M, Garrett Z, Rush K, Konečný J, Kumar S, McMahan H B. Adaptive federated optimization. In: Proceedings of the 9th International Conference on Learning Representations. 2021

41.　Liu Q, Yang H, Dou Q, Heng P A. Federated semi-supervised medical image classification via inter-client relation matching. In: Proceedings of the 24th International Conference. 2021, 325−335

42.　Sattler F, Müller K R, Samek W. Clustered federated learning: model-agnostic distributed multitask optimization under privacy constraints. IEEE Transactions on Neural Networks and Learning Systems, 2021, 32(8): 3710−3722

43.　Qayyum A, Ahmad K, Ahsan M A, Al-Fuqaha A, Qadir J. Collaborative federated learning for healthcare: multi-modal covid-19 diagnosis at the edge. IEEE Open Journal of the Computer Society, 2022, 3: 172−184

44.　Mansour Y, Mohri M, Ro J, Suresh A T. Three approaches for personalization with applications to federated learning. 2020, arXiv preprint arXiv: 2002.10619

45.　Huang L, Shea A L, Qian H, Masurkar A, Deng H, Liu D. Patient clustering improves efficiency of federated machine learning to predict mortality and hospital stay time using distributed electronic medical records. Journal of Biomedical Informatics, 2019, 99: 103291

46.　Ouyang X, Xie Z, Zhou J, Xing G, Huang J. ClusterFL: a clustering-based federated learning system for human activity recognition. ACM Transactions on Sensor Networks, 2023, 19(1): 17

47.　Duan M, Liu D, Chen X, Liu R, Tan Y, Liang L. Self-balancing federated learning with global imbalanced data in mobile systems. IEEE Transactions on Parallel and Distributed Systems, 2021, 32(1): 59−71

48.　Zhang X, Mavromatics A, Vafeas A, Nejabati R, Simeonidou D. Federated feature selection for horizontal federated learning in IoT networks. IEEE Internet of Things Journal, 2023, 10(11): 10095−10112

49.　Hu Y, Zhang Y, Gao X, Gong D, Song X, Guo Y, Wang J. A federated feature selection algorithm based on particle swarm optimization under privacy protection. Knowledge-Based Systems, 2023, 260: 110122

50.　Jeong W, Yoon J, Yang E, Hwang S J. Federated semi-supervised learning with inter-client consistency & disjoint learning. In: Proceedings of the 9th International Conference on Learning Representations. 2021

51.　Li S, Zhou T, Tian X, Tao D. Learning to collaborate in decentralized learning of personalized models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022, 9756−9765

52.　Qi T, Wu F, Lyu L, Huang Y, Xie X. FedSampling: a better sampling strategy for federated learning. In: Proceedings of the 32nd International Joint Conference on Artificial Intelligence. 2023, 4154−4162

53.　Chen M, Yang Z, Saad W, Yin C, Poor H V, Cui S. A joint learning and communications framework for federated learning over wireless networks. IEEE Transactions on Wireless Communications, 2021, 20(1): 269−283

54.　Deng Y, Lyu F, Ren J, Wu H, Zhou Y, Zhang Y, Shen X. AUCTION: automated and quality-aware client selection framework for efficient federated learning. IEEE Transactions on Parallel and Distributed Systems, 2022, 33(8): 1996−2009

55.　Yang H H, Arafa A, Quek T Q S, Poor H V. Age-based scheduling policy for federated learning in mobile edge networks. In: Proceedings of 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). 2020, 8743−8747

56.　Su L, Zhou R, Wang N, Fang G, Li Z. An online learning approach for client selection in federated edge learning under budget constraint. In: Proceedings of the 51st International Conference on Parallel Processing. 2022, 72

57.　Wu C, Wu F, Lyu L, Huang Y, Xie X. Communication-efficient federated learning via knowledge distillation. Nature Communications, 2022, 13(1): 2032

58.　Tuor T, Wang S, Ko B J, Liu C, Leung K K. Data selection for federated learning with relevant and irrelevant data at clients. 2020, arXiv preprint arXiv: 2001.08300

59.　Duan M, Liu D, Ji X, Liu R, Liang L, Chen X, Tan Y. FedGroup: Efficient clustered federated learning via decomposed data-driven measure. 2020, arXiv preprint arXiv: 2010.06870

60.　Nagalapatti L, Narayanam R. Game of gradients: mitigating irrelevant clients in federated learning. In: Proceedings of the 35th AAAI Conference on Artificial Intelligence. 2021, 9046−9054

61.　Yoon T, Shin S, Hwang S J, Yang E. FedMix: approximation of Mixup under mean augmented federated learning. In: Proceedings of the 9th International Conference on Learning Representations. 2021

62.　Hao W, El-Khamy M, Lee J, Zhang J, Liang K J, Chen C, Carin L. Towards fair federated learning with zero-shot data augmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). 2021, 3305−3314

63.　Liang P P, Liu T, Liu Z, Allen N B, Auerbach R P, Brent D,

Salakhutdinov R, Morency L P. Think locally, act globally: federated learning with local and global representations. 2020, arXiv preprint arXiv: 2001.01523

64. Briggs C, Fan Z, Andras P. Federated learning with hierarchical clustering of local updates to improve training on non-IID data. In: Proceedings of 2020 International Joint Conference on Neural Networks (IJCNN). 2020, 1−9

65. Liu B, Guo Y, Chen X. PFA: privacy-preserving federated adaptation for effective model personalization. In: Proceedings of the Web Conference 2021. 2021, 923−934

66. Tan Y, Long G, Liu L, Zhou T, Lu Q, Jiang J, Zhang C. FedProto: federated prototype learning across heterogeneous clients. In: Proceedings of the 36th AAAI Conference on Artificial Intelligence. 2022, 8432−8440

67. Shen T, Zhang J, Jia X, Zhang F, Huang G, Zhou P, Kuang K, Wu F, Wu C. Federated mutual learning. 2020, arXiv preprint arXiv: 2006.16765

68. Arivazhagan M G, Aggarwal V, Singh A K, Choudhary S. Federated learning with personalization layers. 2019, arXiv preprint arXiv: 1912.00818

69. Fallah A, Mokhtari A, Ozdaglar A. Personalized federated learning with theoretical guarantees: a model-agnostic meta-learning approach. In: Proceedings of the 34th International Conference on Neural Information Processing Systems. 2020, 300

70. Li T, Sahu A K, Talwalkar A, Smith V. Federated learning: challenges, methods, and future directions. IEEE Signal Processing Magazine, 2020, 37(3): 50−60

71. Deng Y, Kamani M M, Mahdavi M. Adaptive personalized federated learning. 2020, arXiv preprint arXiv: 2003.13461

72. Dinh C T, Tran N H, Nguyen T D. Personalized federated learning with Moreau envelopes. In: Proceedings of the 34th International Conference on Neural Information Processing Systems. 2020, 1796

73. Hanzely F, Richtárik P. Federated learning of a mixture of global and local models. 2020, arXiv preprint arXiv: 2002.05516

74. Dinh C T, Tran N H, Nguyen T D, Bao W, Zomaya A Y, Zhou B B. Federated learning with proximal stochastic variance reduced gradient algorithms. In: Proceedings of the 49th International Conference on Parallel Processing. 2020, 48

75. Hanzely F, Hanzely S, Horváth S, Richtárik P. Lower bounds and optimal algorithms for personalized federated learning. In: Proceedings of the 34th International Conference on Neural Information Processing Systems. 2020, 194

76. Li T, Hu S, Beirami A, Smith V. Ditto: fair and robust federated learning through personalization. In: Proceedings of the 38th International Conference on Machine Learning. 2021, 6357−6368

77. Karimireddy S P, Kale S, Mohri M, Reddi S J, Stich S U, Suresh A T. SCAFFOLD: stochastic controlled averaging for federated learning. In: Proceedings of the 37th International Conference on Machine Learning. 2020, 476

78. Ma Z, Zhao M, Cai X, Jia Z. Fast-convergent federated learning with class-weighted aggregation. Journal of Systems Architecture, 2021, 117: 102125

79. Collins L, Hassani H, Mokhtari A, Shakkottai S. Exploiting shared representations for personalized federated learning. In: Proceedings of the 38th International Conference on Machine Learning. 2021, 2089−2099

80. Oh J, Kim S, Yun S Y. FedBABU: towards enhanced representation for federated image classification. 2021, arXiv preprint arXiv: 2106.06042

81. Jang J, Ha H, Jung D, Yoon S. FedClassAvg: local representation learning for personalized federated learning on heterogeneous neural networks. In: Proceedings of the 51st International Conference on Parallel Processing. 2022, 76

82. Zhuang W, Gan X, Wen Y, Zhang S, Yi S. Collaborative unsupervised visual representation learning from decentralized data. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021, 4892−4901

83. Qu Z, Li X, Han X, Duan R, Shen C, Chen L. How to prevent the poor performance clients for personalized federated learning? In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2023, 12167−12176

84. Wang K, He Q, Chen F, Chen C, Huang F, Jin H, Yang Y. FlexiFed: personalized federated learning for edge clients with heterogeneous model architectures. In: Proceedings of the ACM Web Conference 2023. 2023, 2979−2990

85. Li A, Sun J, Zeng X, Zhang M, Li H, Chen Y. FedMask: joint computation and communication-efficient personalized federated learning via heterogeneous masking. In: Proceedings of the 19th ACM Conference on Embedded Networked Sensor Systems. 2021, 42−55

86. Yang Z, Sun Q. Personalized heterogeneity-aware federated search towards better accuracy and energy efficiency. In: Proceedings of the 41st IEEE/ACM International Conference on Computer Aided Design. 2022, 1−9

87. Diao E, Ding J, Tarokh V. HeteroFL: computation and communication efficient federated learning for heterogeneous clients. In: Proceedings of the 9th International Conference on Learning Representations. 2020

88. Zeng H, Zhou T, Guo Y, Cai Z, Liu F. FedCav: contribution-aware model aggregation on distributed heterogeneous data in federated learning. In: Proceedings of the 50th International Conference on Parallel Processing. 2021, 75

89. Zhu J, Ma X, Blaschko M B. Confidence-aware personalized federated learning via variational expectation maximization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2023, 24542−24551

90. Lin X, Chen H, Xu Y, Xu C, Gui X, Deng Y, Wang Y. Federated learning with positive and unlabeled data. In: Proceedings of the 39th International Conference on Machine Learning. 2022, 13344−13355

91. Beilharz J, Pfitzner B, Schmid R, Geppert P, Arnrich B, Polze A. Implicit model specialization through dag-based decentralized federated learning. In: Proceedings of the 22nd International Middleware Conference. 2021, 310−322

92. Zhang M, Sapra K, Fidler S, Yeung S, Álvarez J M. Personalized federated learning with first order model optimization. In: Proceedings of the 9th International Conference on Learning Representations. 2021

93. Liu J, Wu J, Chen J, Hu M, Zhou Y, Wu D. FedDWA: personalized federated learning with dynamic weight adjustment. In: Proceedings of the 32nd International Joint Conference on Artificial Intelligence. 2023, 3993−4001

94. Wang H, Kaplan Z, Niu D, Li B. Optimizing federated learning on Non-IID data with reinforcement learning. In: Proceedings of the IEEE Conference on Computer Communications. 2020, 1698−1707

95. Nishio T, Yonetani R. Client selection for federated learning with heterogeneous resources in mobile edge. In: Proceedings of 2019 IEEE International Conference on Communications (ICC). 2019, 1−7

96. Wang L, Wang W, Li B. CMFL: mitigating communication overhead for federated learning. In: Proceedings of the 39th International Conference on Distributed Computing Systems (ICDCS). 2019, 954−964

97. Xia W, Quek T Q S, Guo K, Wen W, Yang H H, Zhu H. Multi-armed bandit-based client scheduling for federated learning. IEEE Transactions on Wireless Communications, 2020, 19(11): 7108−7123

98. Yang M, Wang X, Zhu H, Wang H, Qian H. Federated learning with class imbalance reduction. In: Proceedings of the 29th European Signal Processing Conference (EUSIPCO). 2021, 2174−2178

99. Chai Z, Ali A, Zawad S, Truex S, Anwar A, Baracaldo N, Zhou Y, Ludwig H, Yan F, Cheng Y. TiFL: a tier-based federated learning system. In: Proceedings of the 29th International Symposium on High-Performance Parallel and Distributed Computing. 2020, 125−136

100. Li L, Duan M, Liu D, Zhang Y, Ren A, Chen X, Tan Y, Wang C. FedSAE: a novel self-adaptive federated learning framework in heterogeneous systems. In: Proceedings of the International Joint Conference on Neural Networks. 2021, 1−10

101. Cox B, Chen L Y, Decouchant J. Aergia: leveraging heterogeneity in federated learning systems. In: Proceedings of the 23rd ACM/IFIP International Middleware Conference. 2022, 107−120

102. Dong J, Wang L, Fang Z, Sun G, Xu S, Wang X, Zhu Q. Federated class-incremental learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022, 10154−10163

103. Makhija D, Han X, Ho N, Ghosh J. Architecture agnostic federated learning for neural networks. In: Proceedings of the 39th International Conference on Machine Learning. 2022, 14860−14870

104. Huang Y, Chu L, Zhou Z, Wang L, Liu J, Pei J, Zhang Y. Personalized cross-silo federated learning on non-IID data. In: Proceedings of the 35th AAAI Conference on Artificial Intelligence. 2021, 7865−7873

105. Zhu S, Qi Q, Zhuang Z, Wang J, Sun H, Liao J. FedNKD: a dependable federated learning using fine-tuned random noise and knowledge distillation. In: Proceedings of 2022 International Conference on Multimedia Retrieval. 2022, 185−193

106. Chai Z, Chen Y, Anwar A, Zhao L, Cheng Y, Rangwala H. FedAT: a high-performance and communication-efficient federated learning system with asynchronous tiers. In: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis. 2021, 1−17

107. Hu C, Liang H H, Han X M, Liu B A, Cheng D Z, Wang D. Spread: decentralized model aggregation for scalable federated learning. In: Proceedings of the 51st International Conference on Parallel Processing. 2022, 75

108. Zhang X, Li Y, Li W, Guo K, Shao Y. Personalized federated learning via variational Bayesian inference. In: Proceedings of the 39th International Conference on Machine Learning. 2022, 26293−26310

109. Nguyen N H, Le Nguyen P, Nguyen T D, Nguyen T T, Nguyen D L, Nguyen T H, Pham H H, Truong T N. FedDRL: deep reinforcement learning-based adaptive aggregation for non-IID data in federated learning. In: Proceedings of the 51st International Conference on Parallel Processing. 2022, 73

110. Yoon J, Jeong W, Lee G, Yang E, Hwang S J. Federated continual learning with weighted inter-client transfer. In: Proceedings of the 38th International Conference on Machine Learning. 2021, 12073−12086

111. Qu Z, Duan R, Chen L, Xu J, Lu Z, Liu Y. Context-Aware online client selection for hierarchical federated learning. IEEE Transactions on Parallel and Distributed Systems, 2022, 33(12): 4353−4367

112. Marfoq O, Neglia G, Vidal R, Kameni L. Personalized federated learning through local memorization. In: Proceedings of the 39th International Conference on Machine Learning. 2022, 15070−15092

113. Huang W, Ye M, Du B, Gao X. Few-shot model agnostic federated learning. In: Proceedings of the 30th ACM International Conference on Multimedia. 2022, 7309−7316

114. Itahara S, Nishio T, Koda Y, Morikura M, Yamamoto K. Distillation-based semi-supervised federated learning for communication-efficient collaborative training with non-IID private data. IEEE Transactions on Mobile Computing, 2023, 22(1): 191−205

115. Zhang J, Guo S, Guo J, Zeng D, Zhou J, Zomaya A Y. Towards data-independent knowledge transfer in model-heterogeneous federated learning. IEEE Transactions on Computers, 2023, 72(10): 2888−2901

116. Zhuang W, Wen Y, Zhang S. Divergence-aware federated self-supervised learning. In: Proceedings of the 10th ACM International Conference on Multimedia. 2022

117. Wang Y, Xu H, Ali W, Li M, Zhou X, Shao J. FedFTHA: a fine-tuning and head aggregation method in federated learning. IEEE Internet of Things Journal, 2023, 10(14): 12749−12762

118. Gong X, Sharma A, Karanam S, Wu Z, Chen T, Doermann D, Innanje A. Preserving privacy in federated learning with ensemble cross-domain knowledge distillation. In: Proceedings of the 36th AAAI Conference on Artificial Intelligence. 2022, 11891−11899

119. Lin T, Kong L, Stich S U, Jaggi M. Ensemble distillation for robust model fusion in federated learning. In: Proceedings of the 34th International Conference on Neural Information Processing Systems. 2020, 198

120. Lin H, Lou J, Xiong L, Shahabi C. SemiFed: semi-supervised federated learning with consistency and pseudo-labeling. 2021, arXiv preprint arXiv: 2108.09412

121. Lubana E S, Tang C I, Kawsar F, Dick R P, Mathur A. Orchestra: unsupervised federated learning via globally consistent clustering. In: Proceedings of the 39th International Conference on Machine Learning. 2022, 14461−14484

122. Liang X, Lin Y, Fu H, Zhu L, Li X. RSCfed: random sampling consensus federated semi-supervised learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022, 10144−10153

123. Li M, Li Q, Wang Y. Class balanced adaptive pseudo labeling for federated semi-supervised learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2023, 16292−16301

124. Zhang L, Wu D, Yuan X. FedZKT: zero-shot knowledge transfer towards resource-constrained federated learning with heterogeneous on-device models. In: Proceedings of the 42nd International Conference on Distributed Computing Systems (ICDCS). 2022, 928−938

125. Zhou T, Konukoglu E. FedFA: federated feature augmentation. In: Proceedings of the 11th International Conference on Learning Representations. 2023

126. Yue K, Jin R, Pilgrim R, Wong C W, Baron D, Dai H. Neural tangent kernel empowered federated learning. In: Proceedings of the 39th International Conference on Machine Learning. 2022, 25783−25803

127. Long G, Xie M, Shen T, Zhou T, Wang X, Jiang J. Multi-center federated learning: clients clustering for better personalization. World Wide Web, 2023, 26(1): 481−500

128. Li Q, He B, Song D. Model-contrastive federated learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021, 10708−10717

129. Chen H Y, Chao W L. On bridging generic and personalized federated learning for image classification. In: Proceedings of the 10th International Conference on Learning Representations. 2022

130. Yao X, Sun L. Continual local training for better initialization of federated models. In: Proceedings of 2020 IEEE International Conference on Image Processing (ICIP). 2020, 1736−1740

131. Wang J, Liu Q, Liang H, Joshi G, Poor H V. Tackling the objective inconsistency problem in heterogeneous federated optimization. In: Proceedings of the 34th International Conference on Neural Information Processing Systems. 2020, 638

132. Yu S, Nguyen P, Abebe W, Qian W, Anwar A, Jannesari A. SPATL: salient parameter aggregation and transfer learning for heterogeneous federated learning. In: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis. 2022, 1−14

133. Liu C, Yang Y, Cai X, Ding Y, Lu H. Completely heterogeneous federated learning. 2022, arXiv preprint arXiv: 2210.15865

134. Ilhan F, Su G, Liu L. ScaleFL: Resource-adaptive federated learning with heterogeneous clients. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2023, 24532−24541

135. Liu C, Lou C, Wang R, Xi A Y, Shen L, Yan J. Deep neural network fusion via graph matching with applications to model ensemble and federated learning. In: Proceedings of the 39th International Conference on Machine Learning. 2022, 13857−13869

136. Cho Y J, Wang J, Joshi G. Client selection in federated learning: convergence analysis and power-of-choice selection strategies. 2020, arXiv preprint arXiv: 2010.01243

137. Huang T, Lin W, Wu W, He L, Li K, Zomaya A Y. An efficiency-boosting client selection scheme for federated learning with fairness guarantee. IEEE Transactions on Parallel and Distributed Systems, 2021, 32(7): 1552−1564

138. Wang H, Li Y, Xu W, Li R, Zhan Y, Zeng Z. DaFKD: domain-aware federated knowledge distillation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2023, 20412−20421

139. Gong X, Song L, Vedula R, Sharma A, Zheng M, Planche B, Innanje A, Chen T, Yuan J, Doermann D, Wu Z Y. Federated learning with privacy-preserving ensemble attention distillation. IEEE Transactions on Medical Imaging, 2023, 42(7): 2057−2067

140. Li Q, He B, Song D. Practical one-shot federated learning for cross-silo setting. In: Proceedings of the 30th International Joint Conference on Artificial Intelligence. 2021, 1484−1490

141. Sattler F, Marban A, Rischke R, Samek W. Communication-efficient federated distillation. 2020, arXiv preprint arXiv: 2012.00632

142. Zhu Z, Hong J, Zhou J. Data-free knowledge distillation for heterogeneous federated learning. In: Proceedings of the 38th International Conference on Machine Learning. 2021, 12878−12889

143. Zhang L, Shen L, Ding L, Tao D, Duan L Y. Fine-tuning global model via data-free knowledge distillation for non-IID federated learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022, 10164−10173

144. Yang Y, Yang R, Peng H, Li Y, Li T, Liao Y, Zhou P. FedACK: federated adversarial contrastive knowledge distillation for cross-lingual and cross-model social bot detection. In: Proceedings of the ACM Web Conference 2023. 2023, 1314−1323

145. Li G, Hu Y, Zhang M, Liu J, Yin Q, Peng Y, Dou D. FedHiSyn: a hierarchical synchronous federated learning framework for resource and data heterogeneity. In: Proceedings of the 51st International Conference on Parallel Processing. 2022, 8

146. Han S, Park S, Wu F, Kim S, Wu C, Xie X, Cha M. FedX: unsupervised federated learning with cross knowledge distillation. In: Proceedings of the 17th European Conference on Computer Vision. 2022, 691−707

147. Li C, Zeng X, Zhang M, Cao Z. PyramidFL: a fine-grained client selection framework for efficient federated learning. In: Proceedings of the 28th Annual International Conference on Mobile Computing and Networking. 2022, 158−171

148. Zhang S, Li Z, Chen Q, Zheng W, Leng J, Guo M. Dubhe: towards data unbiasedness with homomorphic encryption in federated learning client selection. In: Proceedings of the 50th International Conference on Parallel Processing. 2021, 83

149. Chen H, Frikha A, Krompass D, Gu J, Tresp V. FRAug: tackling federated learning with non-IID features via representation augmentation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. 2023, 4826−4836

150. Liu Q, Chen C, Qin J, Dou Q, Heng P A. FedDG: federated domain generalization on medical image segmentation via episodic learning in continuous frequency space. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021, 1013−1023

151. Li X, Jiang M, Zhang X, Kamp M, Dou Q. FedBN: federated learning on non-IID features via local batch normalization. In: Proceedings of the 9th International Conference on Learning Representations. 2021

152. Yang D, Xu Z, Li W, Myronenko A, Roth H R, Harmon S, Xu S, Turkbey B, Turkbey E, Wang X, Zhu W, Carrafiello G, Patella F, Cariati M, Obinata H, Mori H, Tamura K, An P, Wood B J, Xu D. Federated semi-supervised learning for COVID region segmentation in chest CT using multi-national data from China, Italy, Japan. Medical Image Analysis, 2021, 70: 101992

153. Wang H, Zhao H, Wang Y, Yu T, Gu J, Gao J. FedKC: federated knowledge composition for multilingual natural language understanding. In: Proceedings of the ACM Web Conference 2022. 2022, 1839−1850

154. Wang K, Mathews R, Kiddon C, Eichner H, Beaufays F, Ramage D. Federated evaluation of on-device personalization. 2019, arXiv preprint arXiv: 1910.10252

155. Pillutla K, Malik K, Mohamed A, Rabbat M G, Sanjabi M, Xiao L. Federated learning with partial model personalization. In: Proceedings of the 39th International Conference on Machine Learning. 2022, 17716−17758

156. Li A, Sun J, Li P, Pu Y, Li H, Chen Y. Hermes: an efficient federated learning framework for heterogeneous mobile clients. In: Proceedings of the 27th Annual International Conference on Mobile Computing and Networking. 2021, 420−437

157. Zhuang W, Wen Y, Zhang S. Joint optimization in edge-cloud continuum for federated unsupervised person re-identification. In: Proceedings of the 29th ACM International Conference on Multimedia. 2021, 433−441

158. Zhang R, Xu Q, Yao J, Zhang Y, Tian Q, Wang Y. Federated domain generalization with generalization adjustment. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2023, 3954−3963

159. Ruan Y, Joe-Wong C. FedSoft: soft clustered federated learning with proximal local updating. In: Proceedings of the 36th AAAI Conference on Artificial Intelligence. 2022, 8124−8131

160. Xie H, Xiong L, Yang C. Federated node classification over graphs with latent link-type heterogeneity. In: Proceedings of the ACM Web Conference 2023. 2023, 556−566

161. Donahue K, Kleinberg J M. Optimality and stability in federated learning: a game-theoretic approach. In: Proceedings of the International Conference on Neural Information Processing Systems. 2021, 1287−1298

162. Dai Z, Low B K H, Jaillet P. Federated Bayesian optimization via Thompson sampling. In: Proceedings of the 34th International Conference on Neural Information Processing Systems. 2020, 812

163. Li D, Wang J. FedMD: Heterogenous federated learning via model distillation. 2019, arXiv preprint arXiv: 1910.03581

164. Li Y, Zhou W, Wang H, Mi H, Hospedales T M. FedH2L: federated learning with model and statistical heterogeneity. 2021, arXiv preprint arXiv: 2101.11296

165. Wu Y, Kang Y, Luo J, He Y, Fan L, Pan R, Yang Q. FedCG: leverage conditional GAN for protecting privacy and maintaining competitive performance in federated learning. In: Proceedings of the 31st International Joint Conference on Artificial Intelligence. 2022, 2334−2340

166. Niu Z, Wang H, Sun H, Ouyang S, Chen Y W, Lin L. MCKD: Mutually collaborative knowledge distillation for federated domain adaptation and generalization. In: Proceedings of 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). 2023, 1−5

167. Huang W, Ye M, Shi Z, Li H, Du B. Rethinking federated learning with domain shift: a prototype view. In: Proceedings of 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2023, 16312−16322

168. Wang H, Yurochkin M, Sun Y, Papailiopoulos D S, Khazaeni Y. Federated learning with matched averaging. In: Proceedings of the 8th International Conference on Learning Representations. 2020

169. Ghosh A, Chung J, Yin D, Ramchandran K. An efficient framework for clustered federated learning. In: Proceedings of the 34th International Conference on Neural Information Processing Systems. 2020, 1643

170. Tu L, Ouyang X, Zhou J, He Y, Xing G. FedDL: federated learning via dynamic layer sharing for human activity recognition. In: Proceedings of the 19th ACM Conference on Embedded Networked Sensor Systems. 2021, 15−28

171. Donahue K, Kleinberg J. Model-sharing games: Analyzing federated learning under voluntary participation. In: Proceedings of the 35th AAAI Conference on Artificial Intelligence. 2021, 5303−5311

172. Gao D, Liu Y, Huang A, Ju C, Yu H, Yang Q. Privacy-preserving heterogeneous federated transfer learning. In: Proceedings of 2019 IEEE International Conference on Big Data (Big Data). 2019, 2552−2559

173. Kang Y, He Y, Luo J, Fan T, Liu Y, Yang Q. Privacy-preserving federated adversarial domain adaptation over feature groups for interpretability. IEEE Transactions on Big Data, 2022, doi: 10.1109/TBDATA.2022.3188292

174. Fu R, Wu Y, Xu Q, Zhang M. FEAST: a communication-efficient federated feature selection framework for relational data. Proceedings of the ACM on Management of Data, 2023, 1(1): 107

175. Banerjee S, Elmroth E, Bhuyan M. Fed-FiS: a novel information-theoretic federated feature selection for learning stability. In: Proceedings of the 28th International Conference on Neural Information Processing. 2021, 480−487

176. Wu Z, Li Q, He B. Practical vertical federated learning with unsupervised representation learning. IEEE Transactions on Big Data, 2022

177. He Y, Kang Y, Zhao X, Luo J, Fan L, Han Y, Yang Q. A hybrid self-supervised learning framework for vertical federated learning. 2022, arXiv preprint arXiv: 2208.08934

178. Feng S, Yu H. Multi-participant multi-class vertical federated learning. 2020, arXiv preprint arXiv: 2001.11154

179. Feng S. Vertical federated learning-based feature selection with non-overlapping sample utilization. Expert Systems with Applications, 2022, 208: 118097

180. Jiang J, Burkhalter L, Fu F, Ding B, Du B, Hithnawi A, Li B, Zhang C. VF-PS: how to select important participants in vertical federated learning, efficiently and securely? In: Proceedings of the 36th International Conference on Neural Information Processing Systems. 2022, 152

181. Castiglia T, Zhou Y, Wang S, Kadhe S, Baracaldo N, Patterson S. LESS-VFL: Communication-efficient feature selection for vertical federated learning. In: Proceedings of the 40th International Conference on Machine Learning. 2023, 3757−3781

182. Kairouz P, McMahan H B, Avent B, Bellet A, Bennis M, et al. Advances and open problems in federated learning. Foundations and Trends® in Machine Learning, 2021, 14(1−2): 1−210

183. Chai Z, Fayyaz H, Fayyaz Z, Anwar A, Zhou Y, Baracaldo N, Ludwig H, Cheng Y. Towards taming the resource and data heterogeneity in federated learning. In: Proceedings of 2019 USENIX Conference on Operational Machine Learning. 2019, 19−21

184. Ye M, Fang X, Du B, Yuen P C, Tao D. Heterogeneous federated learning: state-of-the-art and research challenges. ACM Computing Surveys, 2024, 56(3): 79

185. Schlegel R, Kumar S, Rosnes E, Amat A G I. CodedPaddedFL and CodedSecAgg: Straggler mitigation and secure aggregation in federated learning. IEEE Transactions on Communications, 2023, 71(4): 2013−2027

186. You C, Xiang J, Su K, Zhang X, Dong S, Onofrey J, Staib L, Duncan J S. Incremental learning meets transfer learning: application to multi-site prostate MRI segmentation. In: Proceedings of the 3rd MICCAI Workshop on Distributed, Collaborative, and Federated Learning, and Affordable AI and Healthcare for Resource Diverse Global Health. 2022, 3−16

187. Chen Y, Qin X, Wang J, Yu C, Gao W. FedHealth: a federated transfer learning framework for wearable healthcare. IEEE Intelligent Systems, 2020, 35(4): 83−93

188. Ditzler G, Polikar R. Incremental learning of concept drift from streaming imbalanced data. IEEE Transactions on Knowledge and Data Engineering, 2013, 25(10): 2283−2301

189. Elwell R, Polikar R. Incremental learning of concept drift in nonstationary environments. IEEE Transactions on Neural Networks, 2011, 22(10): 1517−1531

190. Tan D S, Lin Y X, Hua K L. Incremental learning of multi-domain image-to-image translations. IEEE Transactions on Circuits and Systems for Video Technology, 2021, 31(4): 1526−1539

191. Huang Y, Bert C, Gomaa A, Fietkau R, Maier A, Putz F. A survey of incremental transfer learning: combining peer-to-peer federated learning and domain incremental learning for multicenter collaboration. 2023, arXiv preprint arXiv: 2309.17192

192. Tang J, Lin K Y, Li L. Using domain adaptation for incremental SVM classification of drift data. Mathematics, 2022, 10(19): 3579

193. Alam S, Liu L, Yan M, Zhang M. FedRolex: model-heterogeneous federated learning with rolling sub-model extraction. In: Proceedings of the 36th International Conference on Neural Information Processing Systems. 2022, 2152

194. Yu F, Zhang W, Qin Z, Xu Z, Wang D, Liu C, Tian Z, Chen X. Heterogeneous federated learning. 2020, arXiv preprint arXiv: 2008.06767

195. Jin Y, Wei X, Liu Y, Yang Q. Towards utilizing unlabeled data in federated learning: a survey and prospective. 2020, arXiv preprint arXiv: 2002.11545

196. Diao E, Ding J, Tarokh V. SemiFL: semi-supervised federated learning for unlabeled clients with alternate training. In: Proceedings of the 36th International Conference on Neural Information Processing Systems. 2022, 1299

197. Shin J, Li Y, Liu Y, Lee S J. FedBalancer: data and pace control for efficient federated learning on heterogeneous clients. In: Proceedings of the 20th Annual International Conference on Mobile Systems, Applications and Services. 2022, 436−449

198. Pilla L L. Optimal task assignment for heterogeneous federated learning devices. In: Proceedings of 2021 IEEE International Parallel and Distributed Processing Symposium (IPDPS). 2021, 661−670

199. Zhang F, Kuang K, You Z, Shen T, Xiao J, Zhang Y, Wu C, Zhuang Y, Li X. Federated unsupervised representation learning. 2020, arXiv preprint arXiv: 2010.08982

200. Liao Y, Ma L, Zhou B, Zhao X, Xie F. DraftFed: a draft-based personalized federated learning approach for heterogeneous convolutional neural networks. IEEE Transactions on Mobile Computing, 2024, 23(5): 3938−3949

201. Liu Y, Guo S, Zhang J, Zhou Q, Wang Y, Zhao X. Feature correlation-guided knowledge transfer for federated self-supervised learning. 2022, arXiv preprint arXiv: 2211.07364

202. Li T, Sahu A K, Zaheer M, Sanjabi M, Talwalkar A, Smith V. Federated optimization in heterogeneous networks. In: Proceedings of

the Machine Learning and Systems. 2020, 429−450

203. Duan J H, Li W, Zou D, Li R, Lu S. Federated learning with data-agnostic distribution fusion. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2023, 8074−8083

204. Liu X, Xi W, Li W, Xu D, Bai G, Zhao J. Co-MDA: federated multisource domain adaptation on black-box models. IEEE Transactions on Circuits and Systems for Video Technology, 2023, 33(12): 7658−7670

205. Yoon J, Park G, Jeong W, Hwang S J. Bitwidth heterogeneous federated learning with progressive weight dequantization. In: Proceedings of the 39th International Conference on Machine Learning. 2022, 25552−25565

206. Dai Z, Low B K H, Jaillet P. Differentially private federated Bayesian optimization with distributed exploration. In: Proceedings of the International Conference on Neural Information Processing Systems. 2021, 9125−9139

207. Zhu H, Wang X, Jin Y. Federated many-task Bayesian optimization. IEEE Transactions on Evolutionary Computation, 2023

208. Chawla N V, Bowyer K W, Hall L O, Kegelmeyer W P. SMOTE: synthetic minority over-sampling technique. Journal of Artificial Intelligence Research, 2002, 16: 321−357

209. He H, Bai Y, Garcia E A, Li S. ADASYN: Adaptive synthetic sampling approach for imbalanced learning. In: Proceedings of 2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence). 2008, 1322−1328

210. Han H, Wang W Y, Mao B H. Borderline-smote: a new over-sampling method in imbalanced data sets learning. In: Proceedings of the International Conference on Intelligent Computing. 2005, 878−887

211. Kubat M, Matwin S. Addressing the curse of imbalanced training sets: one-sided selection. In: Proceedings of the 14th International Conference on Machine Learning. 1997, 179−186

212. Yen S J, Lee Y S. Cluster-based under-sampling approaches for imbalanced data distributions. Expert Systems with Applications, 2009, 36(3): 5718−5727

213. Tsai C F, Lin W C, Hu Y H, Yao G T. Under-sampling class imbalanced datasets by combining clustering analysis and instance selection. Information Sciences, 2019, 477: 47−54

214. Zhang L, Li Y, Xiao X, Li X Y, Wang J, Zhou A, Li Q. CrowdBuy: privacy-friendly image dataset purchasing via crowdsourcing. In: Proceedings of the IEEE Conference on Computer Communications. 2018, 2735−2743

215. Li A, Zhang L, Qian J, Xiao X, Li X Y, Xie Y. TODQA: efficient task-oriented data quality assessment. In: Proceedings of the 15th International Conference on Mobile Ad-Hoc and Sensor Networks (MSN). 2019, 81−88

216. Katharopoulos A, Fleuret F. Not all samples are created equal: deep learning with importance sampling. In: Proceedings of the 35th International Conference on Machine Learning. 2018, 2530−2539

217. Alain G, Lamb A, Sankar C, Courville A, Bengio Y. Variance reduction in SGD by distributed importance sampling. 2015, arXiv preprint arXiv: 1511.06481

218. Loshchilov I, Hutter F. Online batch selection for faster training of neural networks. 2015, arXiv preprint arXiv: 1511.06343

219. Schaul T, Quan J, Antonoglou I, Silver D. Prioritized experience replay. In: Proceedings of the 4th International Conference on Learning Representations. 2016

220. Wu C Y, Manmatha R, Smola A J, Krahenbuhl P. Sampling matters in deep embedding learning. In: Proceedings of the IEEE International Conference on Computer Vision. 2017, 2859−2867

221. Li K, Xiao C. CBFL: a communication-efficient federated learning framework from data redundancy perspective. IEEE Systems Journal, 2022, 16(4): 5572−5583

222. Duan L, Tsang I W, Xu D, Chua T S. Domain adaptation from multiple sources via auxiliary classifiers. In: Proceedings of the 26th Annual International Conference on Machine Learning. 2009, 289−296

223. Duan L, Xu D, Tsang I W H. Domain adaptation from multiple sources: a domain-dependent regularization approach. IEEE Transactions on Neural Networks and Learning Systems, 2012, 23(3): 504−518

224. Zhuang F, Luo P, Xiong H, Xiong Y, He Q, Shi Z. Cross-domain learning from multiple sources: a consensus regularization perspective. IEEE Transactions on Knowledge and Data Engineering, 2010, 22(12): 1664−1678

225. Luo P, Zhuang F, Xiong H, Xiong Y, He Q. Transfer learning from multiple source domains via consensus regularization. In: Proceedings of the 17th ACM Conference on Information and Knowledge Management. 2008, 103−112

226. Kirkpatrick J, Pascanu R, Rabinowitz N, Veness J, Desjardins G, Rusu A A, Milan K, Quan J, Ramalho T, Grabska-Barwinska A, Hassabis D, Clopath C, Kumaran D, Hadsell R. Overcoming catastrophic forgetting in neural networks. Proceedings of the National Academy of Sciences of the United States of America, 2017, 114(13): 3521−3526

227. Yu H, Zhang N, Deng S, Yuan Z, Jia Y, Chen H. The devil is the classifier: investigating long tail relation classification with decoupling analysis. 2020, arXiv preprint arXiv: 2009.07022

228. Kang B, Xie S, Rohrbach M, Yan Z, Gordo A, Feng J, Kalantidis Y. Decoupling representation and classifier for long-tailed recognition. In: Proceedings of the 8th International Conference on Learning Representations. 2020

229. Yosinski J, Clune J, Bengio Y, Lipson H. How transferable are features in deep neural networks? In: Proceedings of the 27th International Conference on Neural Information Processing Systems. 2014, 3320−3328

230. Devlin J, Chang M W, Lee K, Toutanova K. BERT: pre-training of deep bidirectional transformers for language understanding. In: Proceedings of 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. 2019, 4171−4186

231. Li X, Huang K, Yang W, Wang S, Zhang Z. On the convergence of FedAvg on non-IID data. In: Proceedings of the 8th International Conference on Learning Representations. 2020

232. Yamada Y, Lindenbaum O, Negahban S, Kluger Y. Feature selection using stochastic gates. In: Proceedings of the 37th International Conference on Machine Learning. 2020, 987

233. Zhou K, Yang Y, Qiao Y, Xiang T. Domain generalization with mixstyle. In: Proceedings of the 9th International Conference on Learning Representations. 2021

234. Li Q, Huang J, Hu J, Gong S. Feature-distribution perturbation and calibration for generalized Reid. In: Proceedings of 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). 2024, 2880−2884

235. Huang X, Belongie S. Arbitrary style transfer in real-time with adaptive instance normalization. In: Proceedings of the IEEE International Conference on Computer Vision. 2017, 1510−1519

236. Han B, Yao Q, Yu X, Niu G, Xu M, Hu W, Tsang I W, Sugiyama M. Co-teaching: Robust training of deep neural networks with extremely noisy labels. In: Proceedings of the 32nd International Conference on Neural Information Processing Systems. 2018, 8536−8546

237. Liu Y, Kang Y, Xing C, Chen T, Yang Q. A secure federated transfer learning framework. IEEE Intelligent Systems, 2020, 35(4): 70−82

238. Yang H, He H, Zhang W, Cao X. FedSteg: a federated transfer learning framework for secure image steganalysis. IEEE Transactions

on Network Science and Engineering, 2021, 8(2): 1084−1094

239.   Li Y, Chen C Y, Wasserman W W. Deep feature selection: theory and application to identify enhancers and promoters. Journal of Computational Biology, 2016, 23(5): 322−336

240.   Louizos C, Welling M, Kingma D P. Learning sparse neural networks through L_0 regularization. In: Proceedings of the 6th International Conference on Learning Representations. 2018

241.   Chang H, Shejwalkar V, Shokri R, Houmansadr A. Cronus: robust and heterogeneous collaborative learning with black-box knowledge transfer. 2019, arXiv preprint arXiv: 1912.11279

242.   Van Berlo B, Saeed A, Ozcelebi T. Towards federated unsupervised representation learning. In: Proceedings of the 3rd ACM International Workshop on Edge Systems, Analytics and Networking. 2020, 31−36

243.   Liu J, Zhao P, Zhuang F, Liu Y, Sheng V S, Xu J, Zhou X, Xiong H. Exploiting aesthetic preference in deep cross networks for cross-domain recommendation. In: Proceedings of the Web Conference 2020. 2020, 2768−2774

244.   Li P, Tuzhilin A. DDTCDR: deep dual transfer cross domain recommendation. In: Proceedings of the 13th International Conference on Web Search and Data Mining. 2020, 331−339

245.   Ammad-Ud-Din M, Ivannikova E, Khan S A, Oyomno W, Fu Q, Tan K E, Flanagan A. Federated collaborative filtering for privacy-preserving personalized recommendation system. 2019, arXiv preprint arXiv: 1901.09888

246.   Minto L, Haller M, Livshits B, Haddadi H. Stronger privacy for federated collaborative filtering with implicit feedback. In: Proceedings of the 15th ACM Conference on Recommender Systems. 2021, 342−350

247.   Chai D, Wang L, Chen K, Yang Q. Secure federated matrix factorization. IEEE Intelligent Systems, 2021, 36(5): 11−20

248.   Du Y, Zhou D, Xie Y, Shi J, Gong M. Federated matrix factorization for privacy-preserving recommender systems. Applied Soft Computing, 2021, 111: 107700

249.   Li Z, Ding B, Zhang C, Li N, Zhou J. Federated matrix factorization with privacy guarantee. Proceedings of the VLDB Endowment, 2021, 15(4): 900−913

250.   Wu C, Wu F, Cao Y, Huang Y, Xie X. FedGNN: federated graph neural network for privacy-preserving recommendation. 2021, arXiv preprint arXiv: 2102.04925

251.   Zhang C, Long G, Zhou T, Yan P, Zhang Z, Zhang C, Yang B. Dual personalization on federated recommendation. In: Proceedings of the 32nd International Joint Conference on Artificial Intelligence. 2023, 4558−4566

252.   Wu J, Liu Q, Huang Z, Ning Y, Wang H, Chen E, Yi J, Zhou B. Hierarchical personalized federated learning for user modeling. In: Proceedings of the Web Conference 2021. 2021, 957−968

253.   Wu M, Li L, Chang T, Rigall E, Wang X, Xu C Z. FedCDR: federated cross-domain recommendation for privacy-preserving rating prediction. In: Proceedings of the 31st ACM International Conference on Information & Knowledge Management. 2022, 2179−2188

254.   Luo S, Xiao Y, Song L. Personalized federated recommendation via joint representation learning, user clustering, and model adaptation. In: Proceedings of the 31st ACM International Conference on Information & Knowledge Management. 2022, 4289−4293

255.   Kaissis G A, Makowski M R, Rückert D, Braren R F. Secure, privacy-preserving and federated machine learning in medical imaging. Nature Machine Intelligence, 2020, 2(6): 305−311

256.   Sui D, Chen Y, Zhao J, Jia Y, Xie Y, Sun W. FedED: federated learning via ensemble distillation for medical relation extraction. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). 2020, 2118−2128

257.   Silva S, Gutman B A, Romero E, Thompson P M, Altmann A, Lorenzi M. Federated learning in distributed medical databases: meta-analysis of large-scale subcortical brain data. In: Proceedings of the 16th IEEE international symposium on biomedical imaging (ISBI 2019). 2019, 270−274

258.   Jin H, Dai X, Xiao J, Li B, Li H, Zhang Y. Cross-cluster federated learning and blockchain for internet of medical things. IEEE Internet of Things Journal, 2021, 8(21): 15776−15784

259.   Xia Y, Yang D, Li W, Myronenko A, Xu D, Obinata H, Mori H, An P, Harmon S, Turkbey E, Turkbey B, Wood B, Patella F, Stellato E, Carrafiello G, Ierardi A, Yuille A, Roth H. Auto-FedAvg: learnable federated averaging for multi-institutional medical image segmentation. 2021, arXiv preprint arXiv: 2104.10195

260.   Jiang M, Roth H R, Li W, Yang D, Zhao C, Nath V, Xu D, Dou Q, Xu Z. Fair federated medical image segmentation via client contribution estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2023, 16302−16311

261.   Acar D A E, Zhao Y, Navarro R M, Mattina M, Whatmough P N, Saligrama V. Federated learning based on dynamic regularization. In: Proceedings of the 9th International Conference on Learning Representations. 2021

262.   Chen Z, Yang C, Zhu M, Peng Z, Yuan Y. Personalized retrogress-resilient federated learning toward imbalanced medical data. IEEE Transactions on Medical Imaging, 2022, 41(12): 3663−3674

263.   Xu A, Li W, Guo P, Yang D, Roth H, Hatamizadeh A, Zhao C, Xu D, Huang H, Xu Z. Closing the generalization gap of cross-silo federated medical image segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022, 20834−20843

264.   Jiang M, Yang H, Cheng C, Dou Q. IOP-FL: Inside-outside personalization for federated medical image segmentation. IEEE Transactions on Medical Imaging, 2023, 42(7): 2106−2117

265.   Wang J, Jin Y, Stoyanov D, Wang L. FedDP: dual personalization in federated medical image segmentation. IEEE Transactions on Medical Imaging, 2024, 43(1): 297−308

266.   Adnan M, Kalra S, Cresswell J C, Taylor G W, Tizhoosh H R. Federated learning and differential privacy for medical image analysis. Scientific Reports, 2022, 12(1): 1953

267.   Wu Y, Zeng D, Wang Z, Shi Y, Hu J. Federated contrastive learning for volumetric medical image segmentation. In: Proceedings of the 24th International Conference on Medical Image Computing and Computer Assisted Intervention. 2021, 367−377

268.   Li Y, Wen G. Research and practice of financial credit risk management based on federated learning. Engineering Letters, 2023, 31(1):

269.   Xu Z, Cheng J, Cheng L, Xu X, Bilal M. Mses credit risk assessment model based on federated learning and feature selection. Computers, Materials & Continua, 2023, 75(3): 5573−5595

270.   Lee C M, Delgado Fernández J, Potenciano Menci S, Rieger A, Fridgen G. Federated learning for credit risk assessment. In: Proceedings of the 56th Hawaii International Conference on System Sciences. 2023, 386−395

271.   Yang W, Zhang Y, Ye K, Li L, Xu C Z. FFD: a federated learning based method for credit card fraud detection. In: Proceedings of the 8th International Congress on Big Data. 2019, 18−32

272.   Wang Z, Xiao J, Wang L, Yao J. A novel federated learning approach with knowledge transfer for credit scoring. Decision Support Systems, 2024, 177: 114084

273.   Pourroostaei Ardakani S, Du N, Lin C, Yang J C, Bi Z, Chen L. A federated learning-enabled predictive analysis to forecast stock market trends. Journal of Ambient Intelligence and Humanized Computing, 2023, 14(4): 4529−4535

274.   Yan Y, Yang G, Gao Y, Zang C, Chen J, Wang Q. Multi-participant

vertical federated learning based time series prediction. In: Proceedings of the 8th International Conference on Computing and Artificial Intelligence. 2022, 165−171

275. Shaheen M, Farooq M S, Umer T. Reduction in data imbalance for client-side training in federated learning for the prediction of stock market prices. Journal of Sensor and Actuator Networks, 2024, 13(1): 1

276. Myalil D, Rajan M A, Apte M, Lodha S. Robust collaborative fraudulent transaction detection using federated learning. In: Proceedings of the 2021 20th IEEE International Conference on Machine Learning and Applications (ICMLA). 2021, 373−378

277. Abadi A, Doyle B, Gini F, Guinamard K, Murakonda S K, Liddell J, Mellor P, Murdoch S J, Naseri M, Page H, Theodorakopoulos G, Weller S. Starlit: Privacy-preserving federated learning to enhance financial fraud detection. 2024, arXiv preprint arXiv: 2401.10765

278. Liu Y, Yu J J Q, Kang J, Niyato D, Zhang S. Privacy-preserving traffic flow prediction: a federated learning approach. IEEE Internet of Things Journal, 2020, 7(8): 7751−7763

279. Zhang C, Dang S, Shihada B, Alouini M S. Dual attention-based federated learning for wireless traffic prediction. In: Proceedings of the IEEE Conference on Computer Communications. 2021, 1−10

280. Zeng T, Guo J, Kim K J, Parsons K, Orlik P, Di Cairano S, Saad W. Multi-task federated learning for traffic prediction and its application to route planning. In: Proceedings of 2021 IEEE Intelligent Vehicles Symposium (IV). 2021, 451−457

281. Zhang C, Cui L, Yu S, Yu J J Q. A communication-efficient federated learning scheme for IoT-based traffic forecasting. IEEE Internet of Things Journal, 2022, 9(14): 11918−11931

282. Qi T, Chen L, Li G, Li Y, Wang C. FedAGCN: a traffic flow prediction framework based on federated learning and asynchronous graph convolutional network. Applied Soft Computing, 2023, 138: 110175

283. Phyu H P, Stanica R, Naboulsi D. Multi-slice privacy-aware traffic forecasting at ran level: a scalable federated-learning approach. IEEE Transactions on Network and Service Management, 2023, 20(4): 5038−5052

284. Xia M, Jin D, Chen J. Short-term traffic flow prediction based on graph convolutional networks and federated learning. IEEE Transactions on Intelligent Transportation Systems, 2023, 24(1): 1191−1203

285. Zhang L, Zhang C, Shihada B. Efficient wireless traffic prediction at the edge: a federated meta-learning approach. IEEE Communications Letters, 2022, 26(7): 1573−1577

286. Hu S, Ye Y, Hu Q, Liu X, Cao S, Yang H H, Shen Y, Angeloudis P, Parada L, Wu C. A federated learning-based framework for ride-sourcing traffic demand prediction. IEEE Transactions on Vehicular Technology, 2023, 72(11): 14002−14015

287. Huo J T, XU Y W, Huo Z S, Xiao L M, He Z X. Research on key technologies of edge cache in virtual data space across WAN. Frontiers of Computer Science, 2023, 17(1): 171102.

288. Li H Z, Jin H, Zheng L, Huang Y, Liao X F. ReCSA: a dedicated sort accelerator using ReRAM-based content addressable memory. Frontiers of Computer Science, 2023, 17(2): 172103.

289. Jia J, Liu Y, Zhang G Z, Gao Y L, Qian D P. Software approaches for resilience of high performance computing systems: a survey. Frontiers of Computer Science, 2023, 17(4): 174105.

290. Guo J Y, Zhang L, ROMERO HUNG J, Li C, Zhao J R, Guo M Y. FPGA sharing in the cloud: a comprehensive analysis. Frontiers of Computer Science, 2023, 17(5): 175106.

Wei Guo is a PhD student at the Institute of Artificial Intelligence, Beihang University, China. She received his MSc degree from the School of Electronics and Computer Science at Southampton University, UK. Her research interests primarily lie in federated learning and transfer learning.

Fuzhen Zhuang received the BE degree from the College of Computer Science, Chongqing University, China in 2006, and the PhD degree in computer science from the Institute of Computing Technology, Chinese Academy of Sciences, China in 2011. He is currently a full professor with the Institute of Artificial Intelligence, Beihang University, China. He has published more than 140 papers in some prestigious refereed journals and conference proceedings. His research interests include transfer learning, machine learning, data mining, multitask learning, knowledge graph, and recommendation systems. He is a senior member of the CCF. He was a recipient of the Distinguished Dissertation Award of CAAI in 2013.

Xiao Zhang is an associate professor at the School of Computer Science and Technology, Shandong University, China. His research interests include distributed learning, federated learning, edge intelligence, and data mining. He has published more than 20 papers in prestigious refereed journals and conference proceedings, such as IEEE TKDE, TMC, UBICOMP, SIGKDD, SIGIR, IJCAI, ACM CIKM, and IEEE ICDM.

Yiqi Tong is a PhD student at the School of Computer Science and Engineering, Beihang University, China. He received his MSc degree from the School of Informatics at Xiamen University, China. His research interests primarily lie in natural language processing and recommendation systems.

Jin Dong is the General Director of Beijing Academy of Blockchain and Edge Computing (BABEC), Director of National Blockchain Technology Innovation Center, Director of Beijing Advanced Innovation Center for Future Blockchain and Privacy Computing. He has been dedicated to technical research in the fields of blockchain, privacy computing, chip design, etc. The team led by him developed the first of kind high performance hardware-software integrated blockchain system - ChainMaker around the globe, aiming to break through the performance and security bottlenecks of large-scale blockchain applications. This has been widely adopted by a variety of key economic and industrial applications in China. Jin Dong received his PhD degree from Tsinghua University, China and has filed more than forty US patents.