

# Unleashing the Power of Continual Learning on Non-Centralized Devices: A Survey

Yichen Li, Haozhao Wang, Wenchao Xu, Tianzhe Xiao, Hong Liu, Minzhu Tu, Yuying Wang, Xin Yang, Rui Zhang, *Senior Member, IEEE*, Shui Yu, *Fellow, IEEE*, Song Guo, *Fellow, IEEE*, Ruixuan Li

**Abstract**—Non-Centralized Continual Learning (NCCL) has become an emerging paradigm for enabling distributed devices such as vehicles and servers to handle streaming data from a joint non-stationary environment. To achieve high reliability and scalability in deploying this paradigm in distributed systems, it is essential to overcome challenges stemming from both spatial and temporal dimensions, manifesting as distribution shifts, catastrophic forgetting, heterogeneity, and privacy issues. This survey focuses on a comprehensive examination of the development of the non-centralized continual learning algorithms and the real-world deployment across distributed devices. We begin with an introduction to the background and fundamentals of non-centralized learning and continual learning. Then, we review existing solutions from three levels to represent how existing techniques alleviate the catastrophic forgetting and distribution shift. Additionally, we delve into the various types of heterogeneity issues, security, and privacy attributes, as well as real-world applications across three prevalent scenarios. Furthermore, we establish a large-scale benchmark to revisit this problem and analyze the performance of the state-of-the-art NCCL approaches. Finally, we discuss the important challenges and future research directions in NCCL.

**Index Terms**—Non-Centralized Continual Learning, Catastrophic Forgetting, Heterogeneity, Security and Privacy, Real-World Applications.

## I. INTRODUCTION

### A. Background

Artificial intelligence, especially deep learning, stands to benefit immensely from the vast amount of data expected in the coming years [1]. Many algorithms have proven effective across various industrial, scientific, and engineering applications. However, data generally originates from a massive

Yichen Li, Haozhao Wang, Tianzhe Xiao, Hong Liu, Rui Zhang, and Ruixuan Li are with the School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan, China. E-mail: {ycli0204, hz\_wang, tz\_xiao, lh678}@hust.edu.cn, rayteam@yeah.net, rxli@hust.edu.cn

Wenchao Xu is with the Department of Computing, The Hong Kong Polytechnic University, Hong Kong, China. E-mail: wenchao.xu@polyu.edu.hk

Minzhu Tu is with the School of Computer Science (National Pilot Software Engineering School), Beijing University of Posts and Telecommunications, Beijing, China. E-mail: cipher0.1104@gmail.com

Yuying Wang is with the School of Computer Science and Technology, Soochow University, Suzhou, China. E-mail: yywang23@stu.suda.edu.cn

Xin Yang is with the School of Computing and Artificial Intelligence, Southwestern University of Finance and Economics, Chengdu, China. E-mail: yangxin@swufe.edu.cn

Shui Yu is with the School of Computer Science, University of Technology Sydney, Sydney, Australia. E-mail: Shui.Yu@uts.edu.au

Song Guo is with the Department of Computer Science and Engineering, The Hong Kong University of Science and Technology, Hong Kong, China. E-mail: songguo@cse.ust.hk

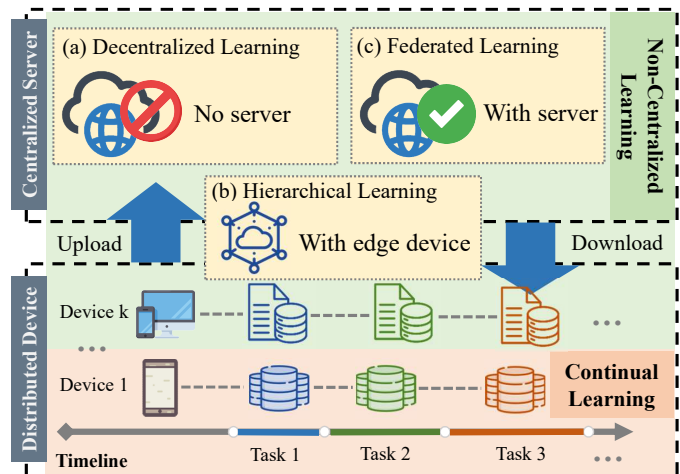


Fig. 1. The overview of NCCL paradigms, including the distributed device layer and centralized server layer with the continual learning process and three different NCL learning paradigms. Each device will collect new data during the training process and collaboratively maintain the model on all streaming tasks without breaching data privacy.

number of devices and is also commonly stored in a distributed manner for a wide range of scenarios. For example, private photos are locally stored on mobile phones, and trajectory data is stored in vehicles. In such cases, data collection in central entities, referred to as centralized training, is often infeasible or impractical due to limited communication resources, data privacy concerns, or country regulations. Currently, the Non-Centralized Learning (NCL) paradigm emerges as a positive response to the increasing need for distributed learning and other concerns in deep learning applications [2], [3]. In NCL, distributed devices process data locally and share knowledge without compromising communication overhead and privacy issues. Among the various NCL algorithms, Federated Learning (FL), introduced by [4] in 2017, has gained wide acknowledgment for its ability to enable entities (also referred to as participants, clients, or parties) to collaboratively train models without sharing sensitive training data. Following years of development, NCL has yielded impressive results, particularly in areas such as edge-cloud collaboration [5], [6], [7], decentralized learning [8], [9], [10], federated learning [11], [12], [13], edge computing [14], [15], [16], and beyond [16]. Meanwhile, it has been applied in various areas such as recommendation systems [17], [18], [19] and autonomous driving [20], [21], [22].

TABLE I

SUMMARY OF RELATED WORKS VERSUS OUR SURVEY. WE IDENTIFY RELEVANT SURVEYS ACROSS THREE CATEGORIES AND SUMMARIZED THEM BASED ON FOUR ASPECTS: PARADIGM, HETEROGENEITY (IN DATA, MODEL, COMPUTATION, AND COMMUNICATION), THREATS, AND APPLICATION.

Category	Year	Ref.	Paradigm		Heterogeneity				Threat		Appli.	Contributions
			NCL	CL	Data	Model	Compu.	Comm.	Privacy	Security		
Non-Centralized Learning	2021	[28]	✓	✗	✓	✗	✗	✗	✓	✗	✓	Highlight various FL architectures and algorithms, emphasizing the need for non-centralized approaches to address privacy and scalability issues.
	2020	[29]	✓	✗	✗	✗	✓	✓	✓	✓	✓	Survey FL in mobile edge networks with resource allocation and communication efficiency and identify research directions and challenges in applications.
	2021	[30]	✓	✗	✗	✗	✗	✓	✓	✓	✓	Review FL for the IoT and discuss the unique challenges like heterogeneity and resources.
	2021	[31]	✓	✗	✓	✗	✗	✗	✓	✓	✗	Provide a taxonomy and comprehensive survey of privacy-preserving FL techniques and contrast different privacy-preserving mechanisms.
	2023	[32]	✓	✗	✓	✗	✗	✓	✓	✓	✓	Introduce decentralized learning and emphasize the benefits of decentralized approaches in improving scalability, robustness, and fault tolerance.
Continual Learning	2021	[33]	✗	✓	✓	✗	✗	✗	✗	✗	✗	Survey CL methods for classification tasks with various CL strategies, including regularization-based, rehearsal-based, and dynamic architectures.
	2022	[34]	✗	✓	✓	✗	✗	✗	✗	✗	✗	Classify incremental learning into three types: CIL, TIL, and DIL and discuss the unique challenges and research directions for each type of incremental learning.
	2022	[35]	✗	✓	✓	✗	✗	✗	✗	✗	✗	Provide a survey and benchmark evaluation of CIL methods for image classification.
Non-Centralized Continual Learning	2023	[36]	✓	✓	✓	✗	✗	✗	✗	✗	✗	Surveys incremental transfer learning, combining P2P FL and DIL for multi-center collaboration and propose a novel framework for NCCL.
	2022	[37]	✓	✓	✓	✗	✗	✗	✗	✗	✗	Discuss the non-IID data and CL processes in FL with potential solutions and research direction.
	2024	[38]	✓	✓	✓	✗	✗	✓	✗	✗	✗	Integrate CL with FL and highlight the benefits of knowledge fusion in improving FCL systems.
<b>Ours</b>	2024	-	✓	✓	✓	✓	✓	✓	✓	✓	✓	A comprehensive and advanced tutorial on the NCCL with all aspects.

## B. Motivation

Despite its impressive advantages, the traditional NCL paradigm makes an unrealistic assumption: all local data on distributed devices should be known beforehand and will remain static and fixed, contradicting the dynamic nature of the real world. In a realistic NCL application, each client may continue collecting new streaming data and train new data on a model that has converged on previous tasks. The lack of dynamism in NCL settings further exacerbates the problem of Catastrophic Forgetting (CF), which is a major challenge in the centralized Continual Learning (CL) paradigm [23]. As models adapt to newly collected data, they often struggle to retain the knowledge gained from previous tasks, leading to a significant degradation in model performance, especially on tasks learned earlier in the training process [24], [25]. This challenge is more severe in NCL settings, where the data in each device remains inaccessible to others, and devices often lack enough resources to implement algorithms designed for centralized settings. Considering the streaming data in distributed devices over time, the challenge of data distribution shift will deserve more attention. In traditional NCL tasks, each device often encounters heterogeneous data across clients participating in the non-centralized training process [26], [27]. Such inconsistencies are primarily due to diverse data sources, distinct data collection methods, and varying data quality. However, when local training data arrives in a streaming manner, each device should focus on the distribution shift between the current task and previous tasks, leading to the problem of knowledge fusion

from different tasks and convergence of the local model.

To tackle these challenges, Non-Centralized Continual Learning (NCCL) breaks the static limitations of traditional non-centralized learning by learning a task sequence on each distributed device. We illustrate this new paradigm in Fig. 1 Compared to traditional NCL training approaches, implementing CL for NCL training at distributed devices features the following advantages.

- *Low Resource Consumption*: Less computational resources and storage are required to learn the new streaming data. For example, instead of training the new data from scratch, devices only continually train the new data with the model converged on previous tasks. As a result, this significantly reduces the computational resources and storage to cache all data of previous tasks.
- *Better Performance*: Continual learning algorithms are designed for streaming tasks with effective skills like data rehearsal and regularization methods. Especially on distributed devices, sometimes the data of new tasks is limited, making it difficult for the model to learn the new features alone. In such a scenario, the CL algorithm can leverage similar knowledge from previous tasks to enhance the performance of the new task.
- *Low Latency*: With CL, distributed devices can be consistently trained and updated. Meanwhile, in the NCL paradigm, real-time decisions, e.g., event decisions, can be made locally at the edge devices. Therefore, the latency is much lower than that when decisions are made

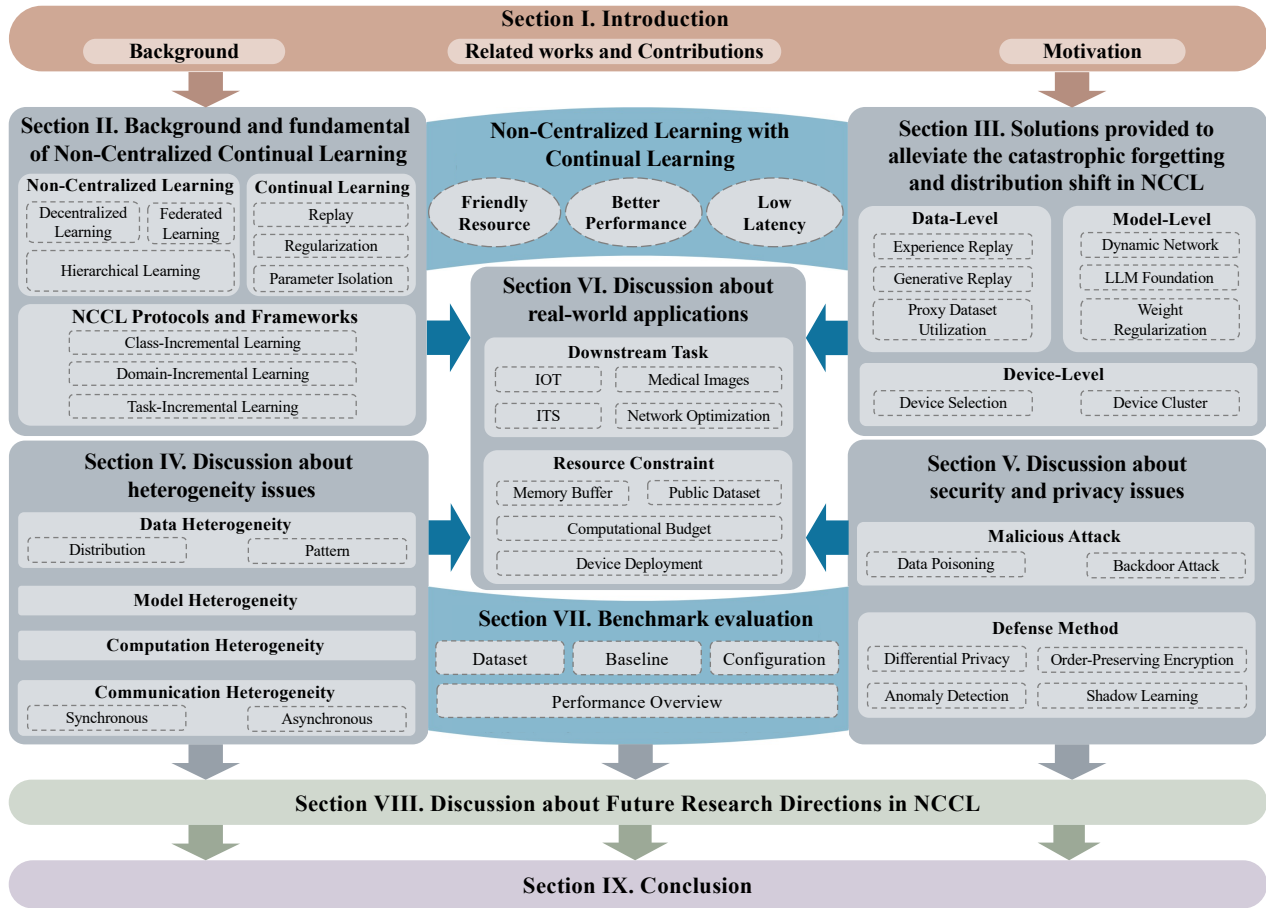


Fig. 2. The outline of this survey, where we introduce the provisioning of continual learning in the non-centralized learning paradigm and highlight some essential issues and challenges.

after the model trains from scratch. This is vital for time-critical applications such as recommendation systems, in which delays can potentially cause users to have a poor experience, thereby reducing economic benefits.

Given the aforementioned advantages, NCCL has seen recent successes in many applications. For instance, federated continual Learning is introduced to tackle the challenge of learning from streaming tasks in individual clients [38], [39]. [40] makes the pioneering effort in this area. Their work focuses on task-incremental learning, requiring distinct task IDs during inference and employing separate task-specific masks to enhance personalized performance. Other studies, such as [41], employ knowledge distillation with a surrogate dataset to retrain the model. Meanwhile, [42], [43], [44] utilizes the generator to synthesize samples to enhance the training process. [45], [46], [47] proposed calculating the importance of samples separately for the local and global distributions, selectively saving necessary samples for retraining to mitigate catastrophic forgetting in federated incremental learning scenarios. Additionally, research like [48], [49] extends NCCL algorithms to medical systems to predict epidemic diseases. In addition, several studies have also explored the use of IoT decentralized learning in some scenarios where a reliable

server is absent, e.g., to foster collaboration across government agencies [50], [51].

Although many research works investigate the integration of CL algorithms with the NCCL paradigm, several challenges must be solved before NCCL can be scaled. Firstly, the NCCL paradigm is relatively complex and diverse, encompassing various sophisticated paradigms such as federated learning, decentralized learning, hierarchical Learning, and more [52], [53]. However, the mainstream of existing research focuses on traditional FL scenarios, which rely on the capabilities of a server, among other factors. Secondly, the distribution of each local streaming data is complex in real-world scenarios. Each time a new task arrives, the distribution of the global data changes. Distributed devices should be able to handle dynamic data distributions, especially when such distributions can be caused by combinations of factors such as sample numbers, label changes, and feature skews [46]. Thirdly, distributed devices like mobile phones are constrained by resources such as storage, computational budget, and sparsely labeled data. The transfer of traditional CL algorithms, which aim to address catastrophic forgetting in the centralized setting, may not work due to insufficient resources [54]. Last but not least, recent research works have clearly shown that a malicious participant

may exist in NCL and can infer the information of other participants just from the shared parameters alone [55]. In particular, centralized CL algorithms do not guarantee privacy in the presence of distributed devices. As such, privacy and security issues in NCCL still need to be considered.

### C. Related Works and Contributions

Although there are surveys on NCL algorithms and CL methods, the existing studies usually treat the two topics separately. For existing surveys on NCL, the authors in [28] place more emphasis on discussing the architecture and categorization of different FL settings to be used for the varying distributions of training data. The authors in [29] highlight the applications of FL in mobile edge networks with resource allocation and communication efficiency. On the other hand, for surveys in CL that focus on implementing CL model training in the centralized setting. For example, the authors in [33], [34] discuss strategies for alleviating catastrophic forgetting and systematically summarize the latest advances in continual learning. Similarly, [35] further provides a benchmark evaluation of existing methods for image classification.

In addition, there are a few existing surveys on federated continual learning. [36] explores the integration of peer-to-peer federated learning and domain incremental learning within the context of incremental transfer learning. This survey analyzes existing methodologies, highlights technical challenges such as domain drift detection and cross-institutional trust mechanisms, and outlines promising directions for applications in healthcare, finance, and IoT ecosystems. The focus of [37] is on non-IID data, briefly mentions the process of continual learning in federated learning, and discusses several common scenarios where continuous learning and federated learning can be simply combined. In contrast, the authors in [38] provide a brief tutorial on the knowledge fusion technique in federated continual learning and the challenges related to its implementation but do not consider the issue of complex heterogeneity, privacy & security, and real-world applications at distributing devices. These surveys also face some common problems, such as incomplete literature coverage (lacking many recent articles published in prominent conferences and journals), relatively simple classification methods that merely migrate the categories of traditional CL in a centralized setting, only focusing on the federated learning scenario among the non-centralized learning paradigm, and neglecting the real-world application of NCCL.

In summary, most existing surveys on NCL do not consider the dynamism of the distributed data collection, whereas surveys on CL do not consider the challenges, including privacy, communication, and heterogeneity. Regarding the few surveys that combine federated learning with continual learning, the analysis of NCCL is not comprehensive and leaves many issues that need to be addressed urgently. We summarize some typical related surveys in comparison to our survey in Table I. Our survey comprehensively reviews NCCL-related research up to March 2025 (over 200 relevant publications retrieved via Google Scholar) and incorporates mainstream taxonomy frameworks from NCL/CL research along with typical real-world applications. In this manuscript, we observe that the

scenarios and experiments in NCCL research are diverse, and there is a lack of a comprehensive survey with benchmark evaluation for NCCL research, which hinders future research in this field. This motivates us to produce a comprehensive survey with the following contributions:

- We motivate the importance of NCCL as an important paradigm shift towards enabling collaborative CL model training on distributed devices. Then, we provide a concise tutorial on NCCL implementation and present a list of useful open-source frameworks for readers that pave the way for future research on NCCL and its applications.
- We discuss the unique features of NCCL relative to a centralized deep learning approach and the resulting implementation challenges. We further raise the challenges of heterogeneity, privacy, and application issues in NCCL. For each of these challenges, we present a comprehensive discussion of existing solutions and approaches explored in existing works.
- We formulate the Federated Continual Learning (FCL), which is the most extensively studied scenario within the NCCL paradigm, and revisit this problem with a large-scale benchmark and analyze the performance of the state-of-the-art FCL approaches.
- We discuss the challenges and future research directions of NCCL.

For the reader's convenience, we classify the related research to be discussed in this survey in Fig. 2. Section II introduces the background and fundamentals of NCCL. Section III reviews solutions provided to alleviate catastrophic forgetting. Section IV discusses heterogeneity across multiple devices in NCCL. Section V discusses privacy and security issues. Section VI discusses applications of NCCL in real-world scenarios. Section VII evaluates some typical existing methods with benchmark experiments. Section VIII discusses the challenges and future research directions in NCCL. Section IX concludes the paper. We also present a list of common abbreviations for reference in Table II.

## II. BACKGROUND AND FUNDAMENTAL OF NON-CENTRALIZED CONTINUAL LEARNING

The main objective of this section is to establish a solid foundation for the reader by providing the background and fundamentals of NCCL. We will first analyze three different NCL paradigms and their challenges, then followed by three distinct CL methods and their respective challenges. Finally, we will present three NCCL protocols and discuss the unique challenges and issues associated with NCCL.

### A. Non-Centralized Learning

We have selected typical popular deep learning-based paradigms related to the NCL: decentralized learning, federated learning, and hierarchical learning. We provide the basic processes and common challenges associated with these paradigms. We illustrate three NCL paradigms in Fig. 3.

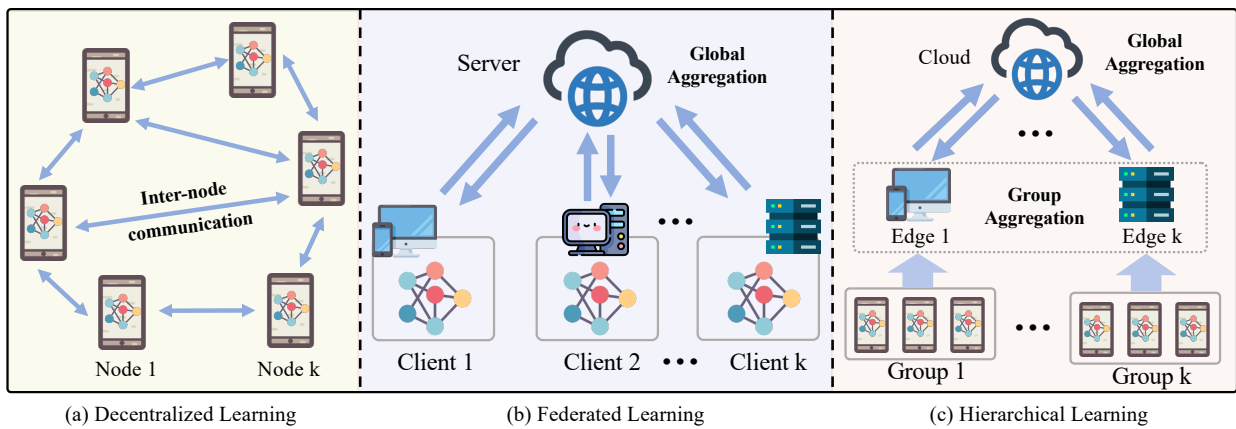


Fig. 3. Three typical Non-Centralized Learning paradigms include (a) Decentralized learning without the central server; (b) Federated learning with a central server aggregating clients; (c) Hierarchical learning with the client-edge-cloud model.

TABLE II  
LIST OF COMMON ABBREVIATIONS IN OUR SURVEY.

Abbreviation	Description
NCL	Non-Centralized Learning
NCCL	Non-Centralized Continual Learning
IID	Independent and Identically Distributed
SGD	Stochastic Gradient Descent
DL	Decentralized Learning
FL	Federated Learning
HL	Hierarchical Learning
CF	Catastrophic Forgetting
IoT	Internet of Things
IoV	Internet of Vehicles
UAV	Unmanned Aerial Vehicle
IL	Incremental Learning
LLM	Large Language Model
RAN	Radio Access Networks
ViT	Vision Transformer
EWC	Elastic Weight Consolidation
LoRA	Low-Rank Adaption
KNN	K-Nearest Neighbors
MMFL	Multi-Modal Federated Learning
GAN	Generative Adversarial Network
LDP	Local Differential Privacy

1) *Decentralized Learning*: Decentralized learning (DL) is a paradigm where central servers are entirely abandoned, leaving only several independent distributed devices [56], [57], [58], [59]. Each device trains the local model with a direct exchange of model information between different devices. This approach brings multiple advantages. Firstly, the high scalability avoids the burden of the central server with a large number of devices. Furthermore, data privacy is also ensured since there is no necessity to send data to the central server. In addition, robustness is preserved when dealing with malfunctions or power outages of devices [60]. At the same time, communication overhead presents a substantial challenge, especially in the involvement of numerous devices.

The DL paradigm typically involves a two-step process: local update and inter-node communication (i.e., communication with neighboring nodes). We usually refer to a distributed

device as a node in DL). Initially, during the local update stage, each node calculates the gradient of its local loss function and updates the model parameters by performing Stochastic Gradient Descent (SGD) multiple times in parallel. When it comes to the inter-node communication stage, each node exchanges its local model parameters with its connected neighbors and then updates its local model  $w_k^{t+1}$  for the next  $(t+1)$ -th round by averaging the received parameters:

$$w_k^{t+1} = \sum_{i \in S_n} c_i w_i^t. \quad (1)$$

where  $S_n$  is the set of neighbors of node  $k$ ,  $c_i$  is the contribution weight assigned to the model of neighboring node  $i$ ,  $w_i^t$  denotes the  $t$ -th round model parameters of node  $i$ .

2) *Federated Learning*: In recent years, with the rising concern of privacy leakage, researchers have focused on finding a trade-off between preserving data privacy and optimizing model training performance. Federated Learning (FL) is proposed for machine learning over distributed local clients, allowing multiple users to contribute to a global model collaboratively by exchanging and aggregating model parameters [61], [62], [63], [54] (we usually refer to a distributed device as a client in FL). Since it involves only the transfer of parameters, the federated learning method significantly reduces communication overhead and effectively preserves data privacy. The foundational federated learning algorithm, FedAvg [64], was introduced by Google in 2017. Since then, numerous studies have been conducted aiming to optimize it further. FedProx [65] significantly alleviates the heterogeneity issue by adding a regularization term to penalize the divergence of the local models from the global model. MOON [66] applies contrastive learning at the model level by comparing model representations to improve the local training of different clients. SCAFFOLD [67] handles the client drift issue caused by non-IID data by introducing control variates to adjust and correct the local updates of each client model.

Then, we will provide a detailed discussion of the federated learning process with three main steps: 1) After determining the training tasks and specifying the hyper-parameters, the server transmits the initialized global model  $w^0$  to all clients;

2) In each communication round  $t$ , a subset of clients is selected randomly. Those participants receive the current global model  $w^t$  and perform SGD on local data. After local training, participating clients will upload the updated parameters to the server; 3) In the aggregation phase, the server aggregates the local models and distributes the updated global model  $w^{t+1}$  to the clients participating in the next round. Particularly, FedAvg uses a weighted mean to balance the contributions based on the number of local data samples, ensuring that clients with more data have a greater influence on the global model:

$$w^{t+1} = \sum_{k \in S_t} \frac{|D_k|}{|D|} w_k^{t+1}. \quad (2)$$

where  $|D|$  is the total number of data,  $|D_k|$  is the number of data on the client  $k$  and  $S_t$  is the set of clients participating in the  $t$ -th round.

3) *Hierarchical Learning*: Hierarchical Learning (HL) is an advanced paradigm of NCL that enhances both scalability and robustness for edge computing systems [68], [69], [70], [71]. It is structured in a client-edge-cloud framework with improved techniques at the edge level to optimize communication and computation costs (we usually refer to a distributed device as a client and construct edge devices and a cloud server in HL).

The main paradigms of HL leverage hierarchical architectures to improve data privacy and system efficiency by forming groups of clients. Specifically, each edge device manages clients and strategically groups them. These groups are then sent to the cloud server, which employs probabilistic group sampling to select some for training. Clients in the participating groups download a global model for local training and send their updated local models back to the edge server for aggregation within the group:

$$w_g^{t,c+1} = \sum_{k \in g} \frac{|D_k|}{|D_g|} w_k^{t,c,e}. \quad (3)$$

Where  $|D_k|$  is the number of data on client  $k$ ,  $|D_g|$  is the number of data for all clients in group  $g$ ,  $w_k^{t,c,e}$  is the local model of client  $k$  after the  $e$ -th local round during the  $c$ -th edge aggregation epoch of the  $t$ -th cloud round.  $w_g^{t,c+1}$  is the group model for group  $g$  in  $(c+1)$ -th edge aggregation round.

After several training rounds, edge servers send the aggregated group models to the cloud for global aggregation. At the end of each global round, the aggregation function is defined as follows, where  $S_t$  denotes the participating client groups,  $w_g^{t,k-1}$  represents the group model at  $(k-1)$ -th group round within  $t$ -th global round, and  $w_g^{t+1}$  is the aggregated global model at the end of global round  $t$ .

$$w_g^{t+1} = \sum_{g \in S_t} \frac{n_g}{m_t} w_g^{t,k-1}. \quad (4)$$

**Challenges of Non-Centralized Learning.** Despite the aforementioned three NCL paradigms serving as a powerful approach that tremendously boosts distributed machine learning, it faces several complex challenges that can hinder general performance and efficiency. Here, we will examine these challenges across three critical aspects: data heterogeneity, data privacy, and resource limitation.

- *Data Heterogeneity*: It is characterized by variances in data types, formats, and distributions on different devices, posing a critical threat to distributed learning and degrading performance in real-world scenarios. Many research studies [72], [73], [74], [75] have contributed to alleviating the non-IID problem through personalization techniques.
- *Data Privacy*: Since transmitting data from edge devices to edge servers still threatens data privacy, one potential solution is to exchange only the parameters of ML models. However, the privacy leakage risk still exists where model parameters may be reversely attacked. A variety of solutions [76], [77], [78], [79] have been proposed to address this issue. For instance, Local Differential Privacy (LDP) is a natural choice, which adds noise without significantly hurting the outcome of any analysis [80].
- *Resource Limitation*: This challenge arises from the constrained computational, communication, or other resources on distributed devices. Limited computational power can slow down training processes, while restricted communication bandwidth can hinder the timely exchange of information between devices and servers. Researchers have explored various strategies [81], [82] to mitigate these resource limitations, such as using lightweight models and reducing communication.

## B. Continual Learning

Continual Learning (CL), also known as lifelong learning [83], [84], [85] or incremental learning [86], [87], refers to the process of acquiring, updating, and applying knowledge over time, much like how humans and other organisms adapt to dynamic environments. It involves learning from a sequential stream of data, where new tasks, domains, or contexts are introduced gradually, while the system aims to retain previously learned knowledge without significant performance degradation, a challenge commonly referred to as catastrophic forgetting. The primary goal of continual learning is to enable AI systems to adapt continuously to changing data and tasks, integrate new knowledge, and improve or maintain performance on earlier tasks without losing previous skills. This is achieved through balancing plasticity and stability, a balance that remains a key challenge in the field. The ultimate aim is to develop systems that can accumulate knowledge over time, transferring what has been learned to better tackle future tasks.

1) *Rehearsal-based method*:: It involves storing a selection of previous data or generating synthetic data for rehearsal. Through the fusion of overall knowledge from both previous and new data, the model performance can be ensured [88], [89]. Given the approach, the loss function is the combination of the new  $t$ -th task loss  $\mathcal{L}_{new}(w^t)$  and a rehearsal loss  $\mathcal{L}_{rehearsal}(w^t)$  which retains information from previous tasks. Here,  $w^t$  denotes the model weights for the  $t$ -th task.  $\alpha$  is a weighting factor that balances the importance.

$$\mathcal{L}(w) = \mathcal{L}_{new}(w^t) + \alpha \cdot \mathcal{L}_{rehearsal}(w^t). \quad (5)$$

2) *Regularization-based method*:: It prevents catastrophic forgetting by adding constraints or regularization terms which

penalizes significant changes to the model parameters that are important for earlier tasks [90], [91], [92]. The total loss function combines the new task loss and a regularization term  $R$  weighted by  $\lambda$  that penalizes deviations of the model  $w^t$  of the  $t$ -th task from their optimal values  $\hat{w}$  of previous tasks. Here,  $i$  represents the index of the previously learned tasks, and  $T - 1$  denotes the total number of these prior tasks.

$$\mathcal{L}(w) = \mathcal{L}_{new}(w^t) + \lambda \sum_{i=1}^{T-1} R(w^t, \hat{w}). \quad (6)$$

3) *Parameter isolation-based method*:: It isolates distinct parameters to specific tasks, which ensures the learning of new tasks does not interfere with the weights and structures learned in the previous tasks [93], [87]. Given the approach, the loss function is:

$$\mathcal{L}(w^t) = \mathcal{L}_{new}(w^t) + \alpha \cdot \mathcal{L}_{isolation}(w^t, \tilde{w}). \quad (7)$$

where  $\mathcal{L}_{isolation}(w^t, \tilde{w})$  denotes the loss measuring differences between new parameters  $w^t$  and isolated parameters  $\tilde{w}$ . **Challenges of Continual Learning.** Despite its ability to handle dynamic streaming data, continual learning (CL) faces several significant challenges: catastrophic forgetting, concept drift, and computational constraints.

- *Catastrophic Forgetting*: A primary issue in CL, where models forget previously learned information upon acquiring new data. This occurs during task training as parameter updates adapt to new data, potentially overwriting crucial parameters for prior tasks, leading to performance degradation. The problem intensifies when new and previous tasks have minimal overlap, as learning the new task often conflicts with retained knowledge. Balancing new and previous knowledge remains a persistent challenge [94], [95], [96].
- *Concept Drift*: Occurs when the statistical properties of the target variable change unexpectedly over time, particularly in the conditional distribution given the input, even if the input distribution remains stable. For instance, consumers' purchasing intentions may evolve while product variety and quantity stay constant. This can prompt the development of drift-aware adaptive learning algorithms to update models online.
- *Computational Constraint*: Frequently encountered in CL, posing a dilemma between high processing demands and limited computing capabilities. To mitigate latency, [97] suggests offloading computations to edge servers with GPUs, shifting the workload from server to edges.

### C. Non-Centralized CL Protocols and Frameworks

NCCL integrates the task scenarios of NCL and CL, aiming to enable each distributed device to dynamically collect and train streaming tasks. Based on the differences among the streaming tasks of the devices, we categorize such scenarios into the following three types [98]. These scenarios are designated as Task-Incremental Learning, Domain-Incremental Learning, and Class-Incremental Learning, and we illustrate it in Fig. 4.

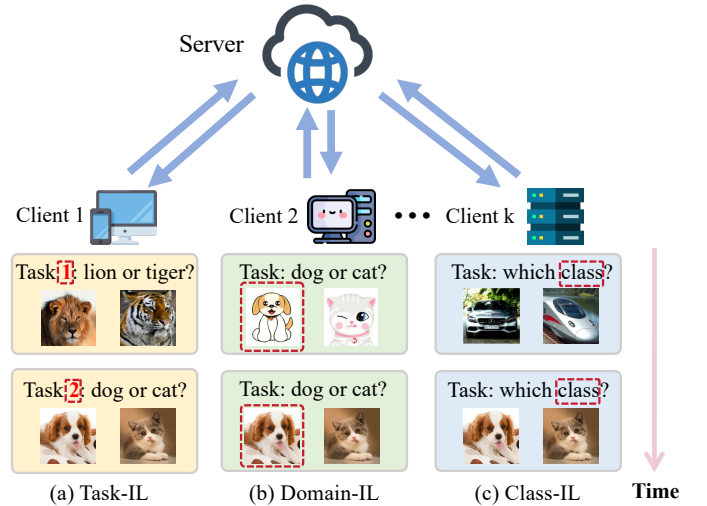


Fig. 4. Three Non-Centralized CL protocols. (a) represents the Task-IL scenario, where the boundaries of different tasks (e.g., task-id) are well-defined; (b) is the Domain-IL scenario, where there exists a feature shift among different tasks, but generally no new classes are introduced; (c) is the Class-IL scenario, where the class types of different tasks do not overlap.

1) *Task-Incremental Learning*: In Task-Incremental Learning (Task-IL), each device progressively acquires knowledge of streaming tasks, with clear boundaries defined for each task. During the inference, the device will know in advance that the sample belongs to the  $t$ -th task. Then, device  $k$  can narrow down the search range of models based on this prior knowledge, thus improving the performance accuracy. However, in the remaining two scenarios, the boundary of each task is ambiguous and task-id is not available during the reference.

2) *Domain-Incremental Learning*: Domain-Incremental Learning (Domain-IL) primarily focuses on changes in input distribution or features over time where the structure of the tasks remains the same. For each task's training dataset  $T_k^t$  at device  $k$ , all tasks in domain incremental learning share the same number of categories. Each task contains all categories, i.e.,  $C_k^t = C_k^1$ , but the data distribution will shift between different tasks, i.e.,  $X_k^t \neq X_k^{t+1}$ .

3) *Class-Incremental Learning*: Class-Incremental Learning (Class-IL) involves the scenario where the device progressively learns to recognize new classes over time. Unlike Task-IL, where the device knows which task a sample belongs to during inference, in Class-IL, the device will classify samples into any of the classes it has encountered so far. Initially, the device is trained on a subset of classes. Subsequently, as new classes are introduced, i.e.,  $C_k^t \neq C_k^1$ , the device updates its model to incorporate this new knowledge without forgetting the previously learned classes.

### D. Unique Challenges and Issues of Non-Centralized CL

Although Non-centralized continual learning can be viewed as integrating CL algorithms into the NCL paradigm, it will introduce the following unique challenges.

- *Local-Global Knowledge Forgetting*: On the one hand, distributed devices locally collect new data, and when

models are trained on this new data, they may overwrite the knowledge learned from old data. On the other hand, data collection efficiency varies between different devices and there are issues with asynchronous communication. Devices that collect data quickly and communicate frequently tend to dominate the knowledge in the global model during global aggregation, potentially overwriting the knowledge from other devices. We summarize this challenge as local-global knowledge forgetting that will significantly impact the performance of NCCL systems.

- *Spatial-Temporal Distribution Shift*: In NCL, different distributed devices often possess different data distributions. This situation evolves further in NCCL, where each distributed device collects new local data during the training process, potentially leading to a shift from its original data distribution. We refer to this phenomenon as the spatial-temporal distribution shift and consider it a significant yet unique challenge in NCCL.
- *Deteriorating Privacy Threats*: Preserving data privacy for distributed devices has always been a paramount challenge in NCL and this issue becomes even more prominent in NCCL due to the need for privacy protection for each task. Furthermore, there may be certain correlations among different tasks, potentially facilitating attackers in obtaining a series of data privacy more conveniently. Meanwhile, to address the aforementioned two challenges, existing methods often employ additional techniques that involve a trade-off between performance and privacy. Therefore, we emphasize here that the issue of privacy and security in NCCL is a challenge that cannot be ignored.
- *Ongoing Resource Overhead*: Resource overhead has been a popular research topic in NCL, as many IoT devices have limited hardware resources, preventing the realization of idealized local training and global communication. However, CL is often challenging and requires sophisticated algorithms and computational resources. Reducing ongoing resource overhead is a necessary challenge for the deployment of NCCL in practical scenarios.

In Section III, we will first delve into existing methods with advanced techniques to alleviate catastrophic forgetting and distribution shift at the data, method, and device levels. Then, the following sections will review related work and address heterogeneity, privacy & security, and real-world application issues separately.

### III. SOLUTIONS PROVIDED FOR ALLEVIATING CATASTROPHIC FORGETTING AND DISTRIBUTION SHIFT IN NON-CENTRALIZED CONTINUAL LEARNING

This section outlines a series of targeted solutions designed to tackle catastrophic forgetting and distribution shifts within NCCL. Considering the research methods of both NCL and CL, we categorize the existing studies into data-level, model-level, and device-level, with 2-3 major techniques.

- *Data-Level Method*: This method primarily centers on the various processing techniques that devices use to manage their local data. A key aspect of this approach

involves the innovative reuse of existing data, which helps to strengthen the model's understanding and retention of previously learned tasks. To achieve this, the method employs strategies such as experience replay, where cached samples from past tasks are revisited [47], or the generation of synthetic data for replay purposes [42]. These techniques ensure that the model remains capable of continuously learning and adapting from an ongoing sequence of tasks. Additionally, the method can incorporate knowledge distillation using proxy datasets [41]. This process further refines the learning experience by extracting and distilling the core knowledge and essential features of the learned tasks into a more compact and efficient model.

- *Model-Level Method*: This category represents the advancement and evolution of learning algorithms in NCCL. These techniques are primarily concerned with refining and enhancing the mechanisms of the model, enabling it to resist forgetting previously learned information naturally. One such technique is weight regularization, which promotes continuity in the learning process by imposing penalties on updates that could disrupt or overwrite the model's memory of prior tasks [99], [100]. Meanwhile, the dynamic network allows the model to remain adaptable to new challenges and tasks while preserving its foundational knowledge. This approach ensures that the model can integrate new information without compromising its existing understanding [101], [102]. Additionally, incorporating Large Language Model (LLM) foundations marks a significant breakthrough in this method. LLMs provide a robust and comprehensive framework for knowledge retention, enabling the model to store and access vast amounts of information in a structured and efficient manner [103].
- *Device-Level Method*: This approach focuses on harnessing the collective intelligence of NCCL through optimizing device participation. Devices possessing complementary datasets work together to foster a comprehensive understanding. By employing meticulous selection criteria and clustering techniques for distributed devices, these strategies augment the capacity for learning and retention [45], [104].

#### A. Data-Level

The data-level methods focus on the strategic use of data to mitigate forgetting. We will explore three different techniques: using cached data from past tasks for replay, synthesizing additional data to assist training, and leveraging external proxy datasets to transfer knowledge between tasks through knowledge distillation. We depict these data-level techniques in Fig. 5 and summarize current main methods in Table III.

1) *Experience Replay*: Its main idea is to use cached data with limited memory buffers to help models retain the knowledge of previous tasks as they learn new ones, which reduces catastrophic forgetting and alleviates distribution shifts. In [121], the authors propose a framework named federated memory strengthening (FedMeS) to address the challenges of

TABLE III  
A SUMMARY OF MAIN DATA-LEVEL METHODS IN NCCL.

Ref	Key Ideas	Advantages	Limitations
<b>Data-Level: Experience Replay</b>			
[105]	Employing a soft data matching strategy to align the on-device data distribution with the cloud-side directory dataset.	Enhanced accuracy, privacy, and reduced costs.	Dependent on data quality and limited adaptability.
[106]	Utilizing experience replay to evaluate distribution shifts in federated medical imaging.	Effective class balancing and pandemic simulation.	Limited attribute diversity and dataset-specific performance.
[107]	Using replay buffers to cache selected samples from the agent’s journey across the network for disaster relief operations.	Significant bandwidth savings and adaptability to disaster scenarios.	Reliant on the data gravity concept, sensitive to node mobility and network conditions.
[108]	Enabling nodes to dynamically update their models with new phishing data streams without data accumulation with replay buffers.	Attention mechanism captures phishing patterns effectively.	Depending on data streams and the replay buffer may cause latency.
[109]	Using data manager component for data loading, partitioning, and distribution across clients, enabling dynamic updates and continual learning in federated continual learning.	Comprehensive vision-centric FL platform that supports diverse tasks and data patterns.	Complexity in customization, potential scalability issues with large-scale deployment.
[110]	Utilizing an uncertainty-aware memory management strategy, which selectively retains and replays critical samples in an online data stream.	Uncertainty-aware memory reduces forgetting in online FCL.	Requires test-time augmentation for Bregman Information estimation.
[111]	Using receiver-initiated data sharing approach, where an agent requests and receives the most relevant data instances from its neighbors to address local knowledge gaps, thereby facilitating incremental learning and adaptation.	Modular parameter sharing enhances performance with low communication cost.	Focuses on supervised learning; lacks reinforcement learning extension.
<b>Data-Level: Generative Replay</b>			
[112]	Leveraging conditional diffusion models to generate synthetic historical data at local devices.	Diffusion models reduce forgetting effectively.	Computationally intensive diffusion training limits scalability.
[113]	Using memory generator model on the central server, which creates pseudo data to mimic the distribution of historical tasks, thereby facilitating knowledge retention across sequential learning tasks.	Central memory rehearsal mitigates catastrophic forgetting efficiently.	Depends heavily on server-side computation.
[114]	Utilizing a self-challenge diffusion model to generate synthetic data by masking and recovering information within samples.	Self-challenge replay and gradient balance address spatiotemporal heterogeneity effectively.	Computational overhead from diffusion models on original data scale.
[115]	Using a feature generator to synthesize features for previous tasks, leveraging knowledge distillation and dynamic adaptive weight allocation	Deconfounded graph transfer learning enhances cross-city scheduling adaptability.	Continual learning’s dynamic nature complicates cross-city transfer.
[116]	Generating pseudo features from prototypes to replace the need for rehearsal memory, thus reducing communication costs and enhancing privacy.	Pseudo feature generation reduces communication costs without rehearsal memory.	Fixed feature extractor and reliance on PCA may limit adaptability and flexibility.
[117]	Employing a local generator model at each client to efficiently pseudo-rehearsal latent features for replay.	Efficient feature aggregation and pseudo-rehearsal.	Generator struggles with complex features.
<b>Data-Level: Proxy Dataset</b>			
[118]	Employing new-class augmented self-distillation in federated class-incremental learning to mitigate catastrophic forgetting by enriching historical model class scores with new class scores predicted by current models.	Distillation harmonizes old and new class scores for effective knowledge transfer.	Distillation relies on accurate historical model outputs.
[119]	Utilizing distillation techniques to mitigate catastrophic forgetting in pervasive computing environments, enabling models to retain the knowledge of previously learned tasks while adapting to new tasks.	Distillation effectively preserves knowledge from past and server models.	Distillation underperforms on balanced tasks and increases complexity.
[120]	Leveraging a proxy dataset to evaluate model updates based on their continuity, thus ensuring robustness against spatiotemporal data shifts.	Enhanced performance with promising privacy.	Limiting its applicability in scenarios with scarce public data.

personalized federated continual learning. FedMeS leverages local memory at each client to store samples from previous tasks, which are then used to both calibrate gradient updates during training and enhance inference via KNN-based Gaussian inference. During the training process, FedMeS uses a novel regularization term that is dynamically adjusted based on the current loss. A dynamic adjustment is employed to help draw useful information from other clients through the global model, facilitating the learning of local tasks. During inference, FedMeS utilizes the local memory to perform KNN-based Gaussian inference. For a test sample, the  $K$  nearest neighbors from the memory are found based on the Euclidean

distance in the feature space. The local prediction is then computed using a Gaussian kernel. The final prediction of FedMeS is obtained by a convex combination of the outputs from the local, personalized model and the KNN inference. Experimental results show that FedMeS significantly outperforms existing baselines in terms of average accuracy and forgetting rate across various datasets and task distributions. For example, on the Split CIFAR-100 dataset with 10 clients, FedMeS achieves an average accuracy of 53.0% and a forgetting rate of 0.06%, compared to the state-of-the-art baselines.

Inspired by [122], [123], the authors in [46] propose Re-Fed to enable each client to cache samples based on their

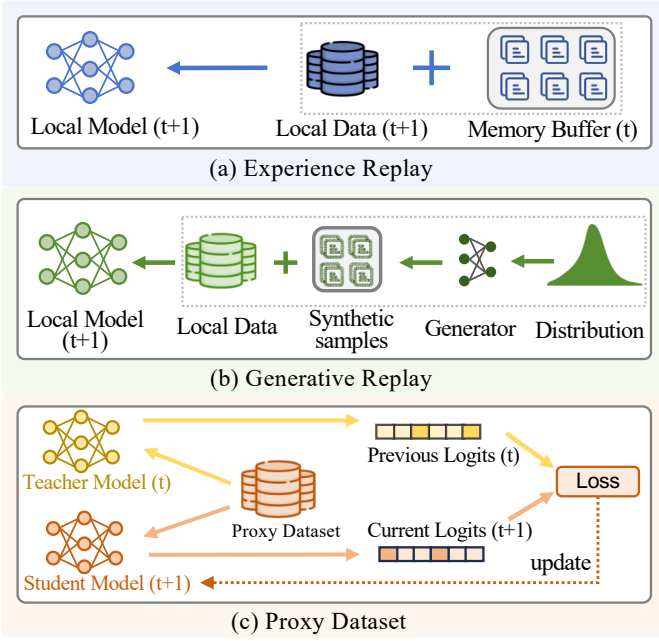


Fig. 5. Data-level methods, including experience replay, generative replay, and proxy dataset. Experience replay focuses on reusing data and features stored in a local buffer; generative replay often relies on generative models to synthesize pseudo-data for replay; and a proxy dataset serves as a medium to facilitate knowledge transfer between different tasks.

calculated importance scores, which are determined by both global and local understanding. This is particularly important given the limited storage capacity at edge clients. Upon the arrival of a new task, each client selects samples for caching using a personalized, informative model. The importance score for each sample is quantified by the gradient norm concerning the personalized informative model parameters. The clients then train their local models with both the cached samples from previous tasks and the samples from the new task, which theoretically helps to alleviate catastrophic forgetting.

Similarly, in [45], the authors aim to address the federated domain-incremental learning with the proposed method SR-FDIL. SR-FDIL is designed to mitigate the effects of catastrophic forgetting by employing a synergistic replay mechanism, allowing clients to cache and replay samples crucial for learning from incremental data with domain shifts. A distinctive feature of SR-FDIL is the use of generative adversarial networks (GAN) to assign a cross-client collaborative score, which is pivotal for determining the global importance of samples. GAN has two main components: a generator and a discriminator. The generator's role is to produce samples that mimic the data distribution, while the discriminator's role is to distinguish between real samples from the data distribution and fake samples generated by the generator. In the context of the SR-FDIL method, the discriminator with parameters is repurposed to assess the global importance of samples rather than distinguishing between real and generated samples. This score is used to determine the significance of the sample in the context of the entire dataset, which is crucial for the sample selection process in SR-FDIL. The local importance is quantified through a domain-representative score, which

measures the similarity of a sample to the local prototype using cosine similarity. The final importance score for each sample is a fusion of both local and global scores, weighted by a hyperparameter. Extensive experiments on various datasets (Digit-10, Office-31, and DomainNet) have been done to illustrate that SR-FDIL can outperform the state-of-the-art methods by up to 4.05% in average accuracy on all domains.

2) *Generative Replay*: It allows each device to synthesize data for replay when the data from previous tasks involves privacy concerns or local storage is insufficient to cache data. In [42], the authors apply ACGAN [124] as the base model and introduce FedCIL, which employs the model consolidation and consistency enforcement of two modules to stabilize the training process and enhance the performance of generative replay in the continual federated learning. On the server side, the model consolidation phase integrates parameters from participating clients to initialize the global model. Subsequently, balanced synthetic data is generated using data provided by the clients. On the client side, consistency enforcement is achieved by aligning the output logits of the classification module with those from the global model and previous local models. This alignment is facilitated through a set of consistency loss functions, which are integrated into the overall loss function. This comprehensive approach ensures that the global model remains robust and capable of classifying all classes learned by the clients thus far without suffering from forgetting.

Based on this, the authors in [43] propose to address the challenge of federated class-continual learning by leveraging knowledge from the global model without requiring real data from previous tasks. On the server side, it uses a generator to synthesize data that conforms to the global model distribution. On the client side, it trains the local model on the real data of the current task and the synthetic data generated for the previous tasks. The training objective combines a cross-entropy loss for the current task and a KL divergence loss for the previous tasks. To further improve the quality of the synthetic data, it employs a student model in addition to the generator. The student model assists in training the generator by providing a distillation loss. This distillation loss helps the generator produce more diverse and valuable synthetic data, enhancing the knowledge transfer from the global model to the local models. Extensive experiments demonstrate the method achieves an accuracy of 36.31%, which is about 6% higher than the best baseline method on CIFAR-100.

Recently, the authors in [44] further propose AF-FCL, which utilizes a normalizing flow (NF) model trained globally to model the feature distribution of previous tasks. The NF model is trained to maximize the likelihood of the input features from client data, as well as sampled features from the previous NF model. During local training, the classifier is optimized using a multi-loss objective consisting of cross-entropy losses on real data, generated data, and a knowledge distillation loss in the feature space. The generated data loss is re-weighted based on the correlation probability of the generated features with the current task distribution.

Similar to [125], [126], [127], authors in [128] proposed hierarchical generative prototypes that are a combination of prompt-based fine-tuning and generative replay. Specifically,

this method constrains the biases in the last layer of the model by using learnable prompts, ensuring that the clients' representations remain close to the pre-training optimum and minimizing federated bias. The server then re-balances the global classifier using generative prototypes. Each client approximates the feature distribution of each class using a multivariate Gaussian distribution. The server identifies the global distribution for each class that minimizes the Jensen-Shannon divergence among all client distributions. The server samples synthetic feature vectors to re-balance the global classifier, effectively mitigating biases in the classification layer. By leveraging prompt-based fine-tuning and hierarchical generative prototypes, the method achieves state-of-the-art performance while maintaining minimal communication costs.

To address catastrophic forgetting in spatiotemporal prediction on streaming data, [129] integrates historical data synthesis, local model training with spatiotemporal mix-up, and federated training to preserve historical knowledge while protecting data privacy. The framework synthesizes historical data using a generator that produces synthetic spatiotemporal data, which is stored in a global replay buffer to simulate the global data distribution without compromising privacy. Each client then fuses current training data with this synthetic data using a spatiotemporal mix-up mechanism, which interpolates between current observations and historical observations sampled from the replay buffer. This interpolation encourages the model to behave linearly across streaming spatiotemporal data sequences, effectively minimizing catastrophic forgetting.

3) *Proxy Dataset*: It assumes that the device or server can acquire other proxy datasets and use them for knowledge distillation to facilitate the transfer of previous and new knowledge. The authors in [118] propose FedCLASS mitigate catastrophic forgetting in federated class-incremental learning scenarios where data volume and class diversity expand over time. The core idea of FedCLASS is to use new-class augmented self-distillation, where class scores of new classes predicted by current models are utilized to enrich the class scores of historical models. This augmented knowledge is then used for self-distillation, enabling more precise knowledge transfer from historical to current models. During the training process, FedCLASS maintains a historical model on each client that represents the model's state before the current incremental task. For each sample, FedCLASS reconstructs the historical model's output by scaling the old class scores and appending new class scores predicted by the current model. This reconstructed output serves as a soft label to guide the training of the current model through self-distillation. Experimental results show that FedCLASS significantly outperforms baseline methods in terms of reducing the average forgetting rate and improving global accuracy. Across four datasets with different class-incremental learning settings, FedCLASS consistently achieves higher global accuracy and lower average forgetting rates compared to state-of-the-art methods. For instance, on the EMNIST dataset with three incremental tasks, FedCLASS achieves a global accuracy of 74.44% and an average forgetting rate of 22.05%, while the best baseline method only reaches 61.79% global accuracy with a 42.43% average forgetting rate.

In [119], the authors introduce two approaches, FLwF-1 and FLwF-2, both of which leverage knowledge distillation to transfer knowledge between models. In FLwF-1, knowledge distillation is employed to transfer knowledge from the past model of a client (teacher) to its current model (student). The distillation loss is computed as the difference between the temperature-scaled logits of the teacher and student models. FLwF-2 extends FLwF-1 by introducing a second teacher model: the current server model. This allows the client model to benefit from the general knowledge encapsulated in the server model and its own past knowledge. The distillation loss for the server model is similarly defined, and the overall loss function for FLwF-2 combines the classification loss, the distillation loss from the past client model, and the distillation loss from the server model. By incorporating knowledge distillation with one or two teacher models, the proposed FLwF approaches aim to mitigate catastrophic forgetting, enabling clients to retain knowledge of previously learned tasks while adapting to new tasks in a continual learning setting.

Similarly, the authors in [130] employ a dual-domain knowledge distillation approach. For the  $t$ -th task, the framework uses two teacher models to guide the student model training. The inter-domain knowledge distillation loss. This loss ensures that the model retains knowledge from previous collaborative training. Additionally, the intra-domain knowledge distillation loss is defined similarly to ensure that the model retains knowledge specific to the local domain. Combining these losses can effectively balance knowledge across different tasks.

## B. Model-Level

These solutions modify the learning process to improve knowledge retention and three effective techniques will be explored in this section. Weight regularization methods, like prior task focus, penalize updates harmful to past tasks, promoting stable task representations. Dynamic Network strategies, such as parameter sharing and freezing, alter model architecture to retain knowledge. Parameter Sharing finds task commonalities, while freezing preserves learned features by restricting updates. Large language models use prompt-based and fine-tuning approaches. Prompt-based methods guide models to integrate cross-task knowledge via designed prompts, enhancing transfer and reducing forgetting. Fine-tuning optimizes retention and adaptation by adjusting key parameters, mitigating forgetting, and protecting privacy. We depict these three model-level methods in Fig. 6 and summarize the currently available main methods in detail in Table IV.

1) *Weight Regularization*: It penalizes weight updates that degrade performance on previously learned tasks [145], [146], anchoring the model to essential knowledge and mitigating catastrophic forgetting. This strategy stabilizes the model's representation by preventing significant changes to weights important for previous tasks. In [147], the authors propose CGoFed, a constrained gradient optimization strategy for federated class incremental learning, which addresses catastrophic forgetting and non-IID label distribution challenges. The core innovation lies in two synergistic modules: relax-constrained gradient update and cross-task gradient regularization. The

TABLE IV  
A SUMMARY OF MAIN MODEL-LEVEL METHODS IN NCCL.

Ref	Key Ideas	Advantages	Limitations
<b>Method-Level: Weight Regularization</b>			
[131]	PI-Fed implements parameter-level importance aggregation to regularize gradients for continual federated learning, mitigating catastrophic forgetting without the need for experience replay.	No Experience Replay Required and Enhancing Efficiency.	Slight increase in communication overhead compared to vanilla FL.
[132]	Leveraging class-specific binary classifiers based on variational auto-encoders to overcome catastrophic forgetting.	Improved efficiency with selective knowledge sharing.	High storage requirement for prototype rehearsal.
[133]	Integrating plasticity and stability in two training stages, employing a task-weighted auxiliary loss for class imbalance and a lightweight distillation method with pre-trained federated models.	Lightweight distillation to reduce communication.	Dependence on pre-trained model architecture.
[134]	Evolving prototypes to transfer knowledge across tasks and constrain local model updates to mitigate catastrophic forgetting.	Effective knowledge retention via evolving prototypes and parameter constraints.	Increased communication and computational overhead.
[135]	Leveraging frozen pre-trained ViT and trainable tail anchors to stabilize feature positions in the spatial-temporal feature space.	Reduced parameter-forgetting and privacy-preserving.	Increased computational cost due to synthetic data generation.
[136]	Combining automatic client weight optimization and dual-domain knowledge distillation to mitigate data heterogeneity and catastrophic forgetting in federated continual learning.	Effective on Non-IID data and improved performance.	Bringing additional potential privacy risks.
[137]	Accumulating global and federated Fisher information matrices to balance parameter importance across tasks and clients, mitigating catastrophic forgetting in federated continual learning.	Robust privacy-preserving with enhanced performance.	Dependence on server validation data.
<b>Method-Level: Dynamic Network</b>			
[138]	Adjusting primary network units and applying weight truncation to enhance model memory and reduce interference.	Multi-PNU architecture for enhanced memory capacity.	High computational complexity due to multiple PNUs.
[139]	Employing group connected and scaling layers to minimize additional parameters for new tasks in VFL.	Reduced parameters compared to initial ones for new tasks.	Complexity from group and scale Layers.
[140]	Decomposing the global model into an adapter for new task learning and a retainer for preserving previous knowledge.	Low storage overhead via knowledge retention.	Dependency on feature similarity for specific datasets.
<b>Method-Level: LLM Foundation</b>			
[103]	Integrating prompt tuning with a fusion function on the client side and prototype-based de-biasing on the server side to address catastrophic forgetting and non-IID data distribution.	Resource-efficient model decomposition.	Complexity of dynamic feature aggregation.
[141]	Utilizing client-side reconstructive prompts and server-side aggregated restoration information to fine-tune classifiers, mitigating forgetting and non-IID effects.	Parameter-efficient learning via prompt tuning.	Dependence on the pre-trained model and large computational overhead.
[142]	Achieving closed-form merging of LoRA modules by alternating between optimizing A and B low-rank matrices, ensuring model response alignment across tasks and clients.	Fast convergence with privacy-preserving.	Dependent on low-rank assumptions and task-specific module storage.
[143]	Employing prompt tuning and hierarchical generative prototypes to constrain and re-balance biases in the classification layer, enhancing model adaptability and communication efficiency.	Efficient communication with privacy-preserving.	Additional computational overhead and Dependence on prototype quality.
[144]	Leveraging cross-task and cross-client knowledge transfer to generate task-specific initializations for efficient adapter fine-tuning through learning attentive weights.	Parameter-efficient and faster learning through informed initialization.	Higher computational overhead and dependence on the historical model.

relax-constrained gradient update mitigates catastrophic forgetting by dynamically restricting gradient updates to subspaces orthogonal to historical task gradients. Unlike strict orthogonal projection, this module introduces a variable constraint strength coefficient that decays exponentially with task progression and adaptively tightens when forgetting exceeds a threshold. This ensures stability for old tasks while preserving plasticity for new tasks. To address non-IID label distributions across clients, the cross-task gradient regularization module leverages historical models from other clients. The server computes a similarity matrix using L2-norm distances between representation matrices and selects relevant historical tasks to construct a regularization loss.

In [148], the authors propose a method called federated orthogonal training that alleviates the catastrophic forgetting problem by ensuring that the updates for the new task are

orthogonal to the global principal component space of activations from the previous tasks at each network layer. The methodology involves maintaining an orthogonal set for each layer, which captures the global principal subspace for all tasks. Similar to [149], when training on a new task, the server projects the aggregated updates onto the orthogonal, effectively minimizing interference and preserving the performance on older tasks. The orthogonal vector set is updated through the global principal subspace extraction step, carried out in an additional communication round after each task. During this round, the server broadcasts the current global model parameters and global principal subspace to all clients. Each client then processes its local data and projects their layer inputs onto the subspace orthogonal. The server then applies singular value decomposition from the matrix formed from these projected inputs to determine the global principal subspace. Empirically,

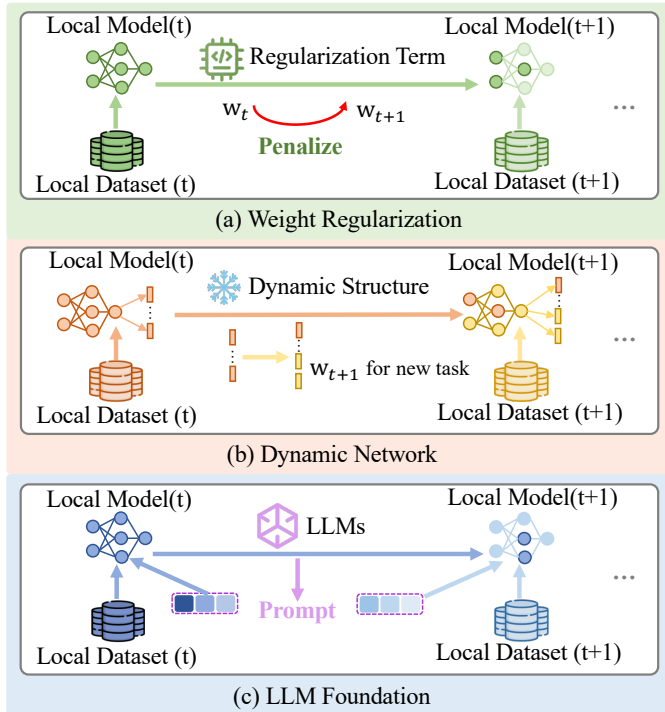


Fig. 6. Model-level methods, including weight regularization, dynamic networks, and LLM-based methods. Regularization-based methods penalize parameter updates through regularization terms, dynamic network-based approaches retain past knowledge by modifying the network structure, and LLM-based methods guide knowledge retention and parameter updates by integrating knowledge from LLM.

in benchmarks such as Permuted MNIST and Split CIFAR-100, the method has shown a 15% increase in average accuracy and a 27% reduction in forgetfulness with minimal increase in computational and communication costs.

In [150], the authors propose FedCurv, an elastic weight merging-based scheme [90] that mitigates the catastrophic forgetting problem in joint learning of non-IID data. EWC prevents catastrophic forgetting by penalizing changes to parameters important for previous task A when learning a new task B. FedCurv applies this idea to federated continual learning by adding a regularization term that encourages all local models to converge to a shared optimum. FedCurv can effectively prevent the model drift of each node and alleviate the catastrophic forgetting problem. Experiments are conducted on the MNIST dataset, simulating a federated continual learning scenario with 96 devices and non-IID data distribution.

2) *Dynamic Network*: The core idea of the dynamic network strategy involves adjusting the model structuring to preserve the acquired knowledge from previous tasks. By implementing parameter sharing for specific model components, the model can distinguish and retain consistent features across various tasks [151], enhancing the generalization of the learned representations. Conversely, parameter freezing strategically keeps partial parameters fixed [152], [153], thereby retaining task-specific features that have been previously learned. This approach to adapting the model's parameter structure ensures proficiency in previously trained tasks while allowing for continuous updates to incorporate new information, effectively

mitigating the issue of catastrophic forgetting.

Similarly, in [154], a strategy based on broad learning is developed as a local-independent training solution, where the local training can be performed independently without relying on the global model's knowledge. The core innovation is introducing a weighted processing strategy that effectively mitigates catastrophic forgetting. This strategy computes a weighted average of the global model weight and the client's local model weight upon receipt from the cloud server. This approach ensures that the updated global model retains previously learned knowledge while adapting to new data. Furthermore, the broad learning technique can effectively support incremental learning by updating the parameters when new data is introduced without retraining the deep architecture. When new data arrives, it enables the model to be incrementally updated without full retraining. The updated feature and enhancement nodes are calculated using the same random weights and biases from the initial setup. Moreover, the broad and flat structure of broad learning networks, with feature and enhancement nodes, enables them to effectively capture the information from new data without relying on deep architectures. This further contributes to the efficiency of the learning process. Empirical evaluations conducted on MNIST and other datasets demonstrate that it not only achieves superior prediction accuracy compared to existing FL schemes but also effectively handles imbalanced and non-IID datasets, outperforming other incremental learning schemes.

In [17], the authors propose a parameter-sharing approach called pFedDIL, which enables each client to adaptively select an appropriate incremental task learning strategy based on the knowledge matching strength and transfer knowledge from previous tasks. To identify the correlations between the new task and previous tasks, pFedDIL employs an auxiliary classifier for each personalized model. The knowledge matching intensity at each client is calculated for each new task by averaging the outputs of the auxiliary classifier across all samples in the new task. Based on this metric, clients can either continue training with a previous model that shares similar knowledge characteristics or start with a newly initialized model for the new task. During training, inspired by ensemble learning [155], each client migrates knowledge from other personalized models using the knowledge matching intensity. Furthermore, pFedDIL shares partial parameters between the target classification model and the auxiliary classifier to condense model parameters. During inference, the final result is obtained by aggregating predictions from all personalized models weighted by the normalized output of the auxiliary classifiers. Experiments on Digit-10, Office-31, and DomainNet datasets demonstrate that pFedDIL outperforms state-of-the-art methods by up to 14.35% in terms of average accuracy across all tasks.

3) *LLM Foundation*: Recently, approaches to LLMs have been introduced in NCCL, which leverage the generalization capabilities of pre-trained models to optimize model adaptation and knowledge retention for previous and new tasks. LLMs such as Vision Transformers (ViT) typically have learned rich feature representations on large-scale datasets during the pre-training phase [156], [157]. In NCCL, these

pre-trained models are used as a basis for fine-tuning a small number of key parameters or directing the model's attention via prompt [158], [103] so that it can recall and utilize the knowledge learned from previous tasks while learning a new task. This strategy not only helps the model remain robust in the face of data distribution shifts [159], but also facilitates knowledge transfer and integration across tasks.

[160] proposes a method using a pre-trained visual transformer ViT combined with the federated multi-granularity prompt to overcome spatial-temporal catastrophic forgetting in personalized federated continual learning. The main idea is to construct a multi-granularity knowledge space by utilizing prompts of different granularities, namely coarse-grained global prompts and fine-grained local prompts, with ViT. It employs ViT as the shared public cognition and trains coarse-grained prompts operating at the input without altering internal parameters to represent temporal-spatial invariant knowledge. The local prompts are built upon the frozen global prompts and directly interact with the model's multi-head self-attention layer. They are designed to capture class-wise fine-grained knowledge for personalization and overcoming temporal forgetting. The local prompt selection is based on the global prompt using a similar key-value pair strategy. Moreover, a selective prompt fusion mechanism is designed to aggregate global prompts from different clients on the server side without spatial forgetting.

Also utilizing ViT and prompt-based approaches, [161] proposed a dual knowledge transfer method to mitigate forgetting, which refers to spatial transfer across different clients and temporal transfer across time-sequential tasks. Prompts are short, trainable sequences concatenated to the input or attention layers of a neural network, allowing the model to adapt to new tasks with minimal parameter changes. The key to preventing forgetting is the selective aggregation of these prompts based on estimated task correlations. The method proposed by authors PKT-FCL introduces a two-step aggregation process for prompt generation: (1) First-step aggregation involves estimating a dual task correlation matrix that captures the similarity between tasks. This matrix is used to selectively aggregate prompts from the global prompt pool based on their relevance to the current task. (2) Second-step aggregation refines this process by selecting only the top-k most relevant tasks for each task, reducing the communication overhead while retaining the essence of knowledge transfer. The dual distillation loss, a cornerstone of the method, ensures that the transferred knowledge is not overwritten while learning new tasks. It adjusts the knowledge distillation process based on the dual-class correlation matrix, which reflects the relevance of classes across tasks. The effectiveness of this method lies in its ability to selectively transfer knowledge that is most relevant to the current task while avoiding the transmission of unnecessary or potentially interfering information. The experiments conducted on the ImageNet-R and DomainNet datasets demonstrate that PKT-FCL significantly enhances accuracy compared to existing methods, showcasing its effectiveness in overcoming catastrophic forgetting in federated continual learning scenarios.

The authors in [162] propose a method called prototype and

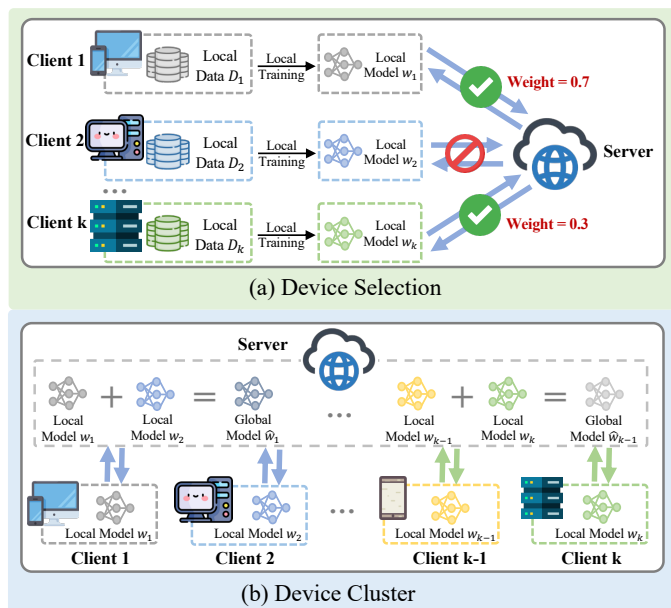


Fig. 7. Device-level methods, including device selection and device cluster. Device selection focuses on assigning different weights to participating devices based on the local data, and the device cluster often aggregates the devices that may share similar local data.

low-rank adaptation (PLoRA) to effectively mitigate the challenges of catastrophic forgetting and data heterogeneity with low communication costs. PLoRA utilizes LoRA to fine-tune the pre-trained transformer model for sequential tasks. Unlike other transformer-based approaches, PLoRA does not rely on module similarity selection mechanisms and is optimized end-to-end. The LoRA module is inserted in the first five blocks of the pre-trained ViT model, and only the parameters of LoRA are trainable and uploaded to the global server during adaptation to new tasks. To address classifier bias caused by data heterogeneity, PLoRA employs prototype learning and designs a prototype re-weight module. In particular, the authors set a prototype for each class. The effectiveness of PLoRA is demonstrated through extensive experiments on CIFAR-100 and Tiny-ImageNet. Moreover, PLoRA exhibits strong robustness and superiority under various non-IID settings and degrees of data heterogeneity.

### C. Device-Level

These methods focus on how to adjust the parameter aggregation strategy based on the data distribution of each device after local training. The device selection approach primarily involves adjusting the weights of different local models within the global model, while the device clustering approach prioritizes aggregating device data with similar data distribution characteristics to mitigate distribution shifts. We illustrate these methods in Table V and Fig. 7.

1) *Device Selection*: It refers to the process of adjusting the weight of participating devices in each training round based on their data and system characteristics [164]. In [45], the authors introduce a balanced client selection mechanism to accelerate convergence and improve model performance during federated domain-incremental learning. This mechanism considers the

TABLE V  
A SUMMARY OF MAIN DEVICE-LEVEL METHODS IN NCCL.

Ref	Key Ideas	Advantages	Limitations
<b>Device-Level: Device Selection</b>			
[45]	Utilizing a balanced client selection mechanism that prioritizes devices based on cache sample importance.	Addressing heterogeneity to improve performance	Data scarcity issues for adversarial training.
[104]	Employing selective device-to-device knowledge transfer to enhance task performance through relevant external knowledge.	Improved accuracy and communication efficiency	Hard to train the model and high training costs.
<b>Device-Level: Device Cluster</b>			
[163]	Utilizing a clustering-based framework with concept matching for federated continual learning that groups client models into concept clusters and collaboratively trains global concept models.	Alleviating catastrophic forgetting and reducing client interference	Increasing client overhead and dependence on concept estimation.

importance of both old and new data by computing an average importance score for each client. During each communication round, each client uploads its score to the server. The server then selects a subset of clients for aggregation through a weighted random sampling method, prioritizing clients with higher scores. This balanced client selection mechanism ensures that clients with more essential data contributions are more likely to be selected, thereby accelerating the learning of new domain samples while maintaining the memory of previous domains.

In [165], each edge device selectively retrieves external knowledge from a neighboring device based on the similarity of their decision patterns. More specifically, after training on a sequence of tasks, a device computes a decision distance vector for its local model. The similarity between two models (the local model and a neighboring model) is then measured using the cosine similarity between their decision distance vectors. Based on this similarity, the device selects the neighboring model with the highest score as the backbone for transferring knowledge to its current task. This approach adjusts the contribution of each device’s knowledge by effectively weighting the devices based on the relevance of their learned representations to the target task. As a result, it enables the system to leverage the most beneficial external knowledge while mitigating the risk of negative transfer, leading to improved model accuracy and more efficient learning in heterogeneous federated continual learning environments.

2) *Device Cluster*: It involves grouping devices based on their model parameters to form multiple centers, each associated with a global model that captures the shared knowledge within that cluster [166]. In [163], concept matching (CM) is proposed to tackle the challenges of catastrophic forgetting and interference among clients in federated continual learning. The core idea is to group client models into concept model clusters and build different global models to capture different concepts over time. At each round, the server sends the global concept models to the clients. To avoid catastrophic forgetting, each client selects the concept model that best matches the concept of the current data for further fine-tuning. This selection is done through a client concept matching algorithm, where the client tests the global concept models on its local data and selects the one with the lowest loss. After receiving the updated client models, the server clusters the

models representing the same concept and aggregates them to form cluster models. Since the server does not know the concepts captured by the aggregated cluster models, it employs a novel server concept matching algorithm to update the global concept models. The server concept matching algorithm uses a distance-based approach to match cluster models with global concept models. For each cluster model, it finds the global concept model with the smallest distance and updates it only if this distance is smaller than the recorded distance from the previous round.

**Conclusion.** In this section, we explore existing approaches addressing two primary challenges in NCCL - catastrophic forgetting and distribution shift - through three key levels: data, model, and device, along with their corresponding techniques. Our analysis reveals that data-level methods have attracted the most research attention. Enhancing models with additional datasets proves to be a relatively effective approach for performance improvement. However, this typically raises privacy leakage concerns and introduces additional training overhead. Model-level methods also demonstrate substantial research activity. While regularization techniques and dynamic network architectures represent extensions of traditional continual learning approaches, recent advancements in LLMs have spurred rapidly emerging research directions that may dominate future studies. Device-level methods remain in their nascent stage, with numerous academic opportunities existing for adjusting client aggregation processes according to streaming tasks in CL scenarios. However, these investigations may pose significant challenges. In practical applications, NCCL itself constitutes an inherently challenging learning paradigm, leading current research to prioritize model performance enhancement over optimizations like lightweight implementations or privacy preservation. Future research directions are recommended as follows: 1) Investigate LLM foundation-based NCCL approaches, focusing on CL optimization methods for LLM base models; 2) Develop context-aware algorithms considering practical constraints such as privacy protection and training environments to improve model applicability; 3) Conduct theoretical analyses of existing NCCL methods to examine convergence properties and interpretability, thereby breaking through current performance limitations.

#### IV. DISCUSSION ABOUT HETEROGENEITY ISSUES

In the aforementioned section, we have analyzed various techniques to address catastrophic forgetting and distribution drift in NCCL. Then, we will delve into the heterogeneity issue, encompassing four common types of heterogeneity in distributed systems: data heterogeneity, model heterogeneity, computation heterogeneity, and communication heterogeneity. These are also the primary issues that traditional NCL aims to resolve. Furthermore, in NCCL, these challenges may be exacerbated because the CL paradigm introduces more complex tasks and higher hardware requirements for training. We illustrate four heterogeneity issues in Fig. 8 and Table VI.

- **Data Heterogeneity:** It is the most prevalent issue in NCL because the simplistic assumption that data on different devices are independent and identically distributed often does not align with reality. This problem persists in NCCL, and as new tasks arrive, it exacerbates the disparity in data distribution among devices. On the one hand, more research uses real-world datasets to conduct experiments [167]. On the other hand, considering the value and privacy concerns of real-world data, most research studies employ mathematical algorithms to partition benchmark datasets to simulate different data distributions [168]. Furthermore, the data pattern is also a crucial aspect, where the label rate and modality on different devices may vary, which can significantly impact the model training paradigm [169].
- **Model Heterogeneity:** It arises when different devices in the distributed system use varying model architectures or parameters. In NCCL, this complexity is heightened due to the continual learning paradigm, where models need to adapt to new tasks without forgetting previous ones [170], [171]. Addressing model heterogeneity involves developing robust model adaptation techniques to align model parameters across devices while maintaining task-specific knowledge. Ensuring consistent model performance across diverse hardware and software configurations is another challenge that researchers must tackle.
- **Computation Heterogeneity:** It refers to the varying computational capabilities among different devices in a distributed system. This type of heterogeneity is particularly relevant in domains such as IoT, airborne systems, and satellite networks. In NCCL, this heterogeneity can lead to uneven workload distribution, where some devices may become bottlenecks due to their limited computational power [172], [173]. To mitigate this issue, researchers must design efficient task allocation and load-balancing strategies. Additionally, leveraging hardware acceleration can help alleviate computational bottlenecks. Ensuring that the computational workload is evenly distributed and that all nodes can contribute effectively to the learning process is crucial for the success of NCCL systems.
- **Communication Heterogeneity:** It stems from the differing communication capabilities and network latencies among devices in a distributed system. In NCCL, this heterogeneity can significantly impact the synchronization of model updates and the overall efficiency of the

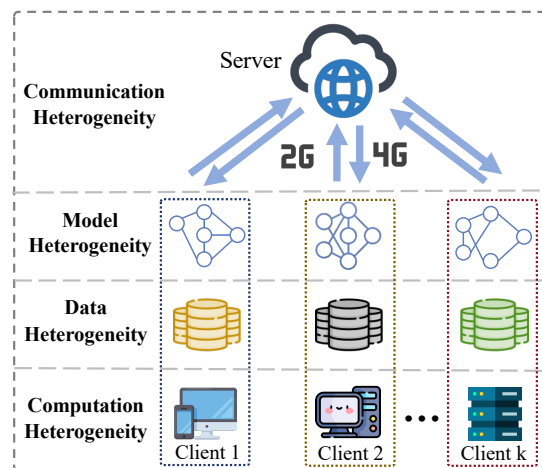


Fig. 8. Heterogeneity issues encompass four types: data, model, computation, and communication. Data heterogeneity focuses on the distribution shift among local datasets. Model heterogeneity refers to the different backbone model architectures on various devices. Computation heterogeneity arises primarily from differences in computing resources due to varying hardware environments. Communication heterogeneity pertains to the differing communication conditions when devices communicate knowledge.

learning process. High network latency or bandwidth limitations can lead to stale gradients and inconsistent model updates [174], [175]. To address this issue, researchers often employ communication-efficient strategies, such as gradient compression and sparse updates, to reduce the amount of data transmitted over the network. Furthermore, developing robust synchronization protocols and leveraging advanced network technologies can help minimize the impact of communication heterogeneity on NCCL systems.

##### A. Data Heterogeneity

Data heterogeneity (i.e., data with different statistical characteristics) has long been a significant issue in model training. Based on existing research, we will summarize here from the perspectives of data distribution and patterns.

1) **Distribution:** We will examine the data partitioning methods previously used in NCCL research to explore heterogeneous data distribution. Dirichlet Distribution is the most commonly used data partitioning method, especially in image classification tasks, where it shows significant advantages [167], [104]. We assume a distribution over  $N$  classes parameterized by a vector  $q(q_i \geq 0, i \in [1, N]$  and  $\|q\|_1 = 1)$ , then  $q \sim \text{Dir}(\alpha p)$  is drawn from a Dirichlet distribution, where  $p$  is a prior class distribution over  $N$  classes and  $\alpha > 0$  is a concentration parameter controlling the identicalness among clients. A smaller parameter  $\alpha$  indicates a more skewed distribution. In [176], the authors divided all data sets into 210 subsets for 7 different clients (30 subsets per client) using the Dirichlet distribution. Specifically, they also discussed the issue of overlapping in partitioning. For example, using the CIFAR-100 dataset, with  $s$  clients and  $m$  subsets per client, when  $s = 285$  and  $m = 30$ , and the local dataset size is 285, the local datasets do not overlap, i.e., 0% overlap. Conversely, when  $s = 213$ , the overlap is 25%, and when  $s = 142$ , the

TABLE VI  
A SUMMARY OF HETEROGENEITY ISSUES IN NCCL.

Ref	Key Ideas	Advantages	Limitations
<b>Heterogeneity Issue: Data Heterogeneity</b>			
[176]	Improving time-evolving federated learning by approximating past objective functions to handle data heterogeneity.	High convergence rate	Information loss
[160]	Utilizing coarse-grained global and fine-grained local prompts to prevent forgetting and achieve effective knowledge personalization.	Efficient knowledge fusion and personalization	Increased computational complexity
[41]	Mitigating catastrophic forgetting by using client-server distillation and surrogate datasets for task review.	Better utilization of computational resources	Relies on surrogate datasets
[112]	Using diffusion models for generating synthetic historical data to mitigate catastrophic forgetting and non-IID challenges.	No need for data storage	Computational overhead
[113]	Utilizing a central server to train a memory generator that creates pseudo data for old tasks, combined with new data to mitigate catastrophic forgetting.	Reduced catastrophic forgetting and privacy preservation	Dependence on synthetic data quality
[177]	Utilizing a gossip protocol and continual learning paradigm to propagate knowledge among clients for naturalistic driving action recognition.	Resource efficiency	Limited generalization
<b>Heterogeneity Issue: Model Heterogeneity</b>			
[178]	Employing experience replay and privacy-preserving generative adversarial networks for decentralized continual learning while maintaining data privacy.	Improved generalization	Increased communication costs
[179]	A federated multi-task learning framework utilizing tensor trace norm regularization to address data, model, and task heterogeneity, enabling efficient knowledge transfer across clients.	Flexibility in handling heterogeneity in data and model	Computational complexity
[170]	Leveraging unlabeled public data to address heterogeneity through cross-correlation learning and alleviate catastrophic forgetting using dual-domain knowledge distillation.	Effective communication across heterogeneous models	Task consistency limitation
[171]	Using federated correlation and similarity learning with non-target distillation to improve both intra-domain discriminability and inter-domain generalization.	Effective cross-domain knowledge transfer	Optimization conflicts in local updating
<b>Heterogeneity Issue: Computational Heterogeneity</b>			
[172]	Integrating signature task knowledge at the client side to mitigate catastrophic forgetting and negative knowledge transfer.	Efficient local computation	Increased complexity with many tasks
[173]	Optimizing computing resource allocation, balancing age-of-information and energy efficiency in dynamic satellite-edge networks.	Dynamic adaptation to heterogeneous environments	Increased complexity in knowledge management
<b>Heterogeneity Issue: Communication Heterogeneity</b>			
[180]	Mitigating catastrophic forgetting through class-aware gradient compensation, class-semantic relation distillation, and a proxy server for model selection.	Dual forgetting compensation	Memory overhead
[175]	Using fractal pre-training, prototype-based learning, contrastive loss, and modified aggregation to handle asynchronous and continual learning across clients.	Independent learning schedules	Scalability challenges
[174]	Utilizing client-side feature translation and server-side prototypical knowledge fusion for spatial-temporal knowledge transfer.	Efficient communication via prototypical knowledge fusion	Computational complexity

overlap is 50%. Although the overlap degree did not affect the performance of different algorithms in the experiments, the authors believe that the overlap parameter and new arriving data will impact the final performance, and they look forward to resolving these issues in future research.

Based on this, the Dirichlet distribution can be combined with other algorithms. In [175], the number of data points per class for each client is computed using a power-law distribution, simulating the imbalance often found in real-world data. For each client, the partitioning is further refined based on a Dirichlet distribution, where the proportions of samples from each class are randomly sampled. The parameter  $\alpha$  of the Dirichlet distribution controls the degree of concentration

Similar to [41], [113], the authors in [112] sort data by categories and then partition them into shards. The data is first sorted according to the labels and then divided into equal-

sized shards, where the number of shards is twice the number of clients. After that, each client is randomly given 2 shards in a non-replicative manner. In the end, each client has data from at most two classes. Furthermore, within each local client, the data is further partitioned into real data and synthetic data generated by the diffusion model. Synthetic data is generated to mitigate the issue of catastrophic forgetting by recovering data distributions. This partitioning allows the local model to train on a mixture of real and synthetic data, improving its robustness to data distribution shifts.

Unlike other research with data partition algorithms, the authors in [177] utilize two real-world datasets for naturalistic driving action recognition: the State Farm Distracted Driver Detection dataset [181] and the Track3NDAR dataset from the 2023 AICity Challenge. Statistical heterogeneity depends on factors such as the driver's physical characteristics, behaviors,

and postures. Within the same cluster, StateFarm clients' data distribution overlaps more, while AICity clients' data distribution is dispersed. For both datasets, a double-splitting approach was employed to divide the data into training and test sets. First, the clients were partitioned into training and test clients with an 80-20 ratio. Secondly, for each client, the local dataset was randomly split into training and test datasets with the same 80-20 ratio. This method ensured that the test data remained unseen during the iterative training process, allowing for an unbiased evaluation of the model's performance on new data. This data partitioning strategy facilitated evaluating the proposed FedPC framework's ability to generalize to new clients and unseen data, which is crucial for assessing its real-world applicability.

2) *Pattern*: Apart from differences in data distribution, the heterogeneity of data patterns also poses significant challenges to the NCCL system. In centralized learning, tasks can be categorized into supervised learning, semi-supervised learning, and unsupervised learning based on the label rate. This issue is inherited and further complicated in NCL and NCCL, as the label rate may vary across each client. According to [169], there are also three different paradigms in federated semi-supervised learning: label-at-all-client-sides, label-at-partial-clients, and label-at-server cases. Therefore, in NCCL, with tasks arriving at different timestamps and exhibiting distinct characteristics, label rates will be even more diverse. In [168], the authors propose a semi-supervised labeling approach to address the issue of partial label data. It utilizes the global model obtained from the cloud to label the unlabeled data in each local device. The confidence of the prediction made by the global model for each unlabeled instance is calculated. This approach allows the local devices to leverage the global knowledge to enrich their local datasets, thus improving the local models over time. To further enhance the robustness against noisy labels that may arise from the semi-supervised labeling process, the paper employs an ensemble-based method for local learning. Each local device maintains an ensemble of base classifiers, and the final prediction is made based on the median of the posterior probabilities provided by these classifiers. The median rule is chosen due to its robustness against outliers. This ensemble-based approach helps to mitigate the impact of mislabeled data and improves the overall performance of the local models.

In addition, the global model selection process is designed to filter out local models that perform poorly or provide noisy data. By applying the distributed effective voting technique [182], the server evaluates the significance of each incoming local model based on its performance on a subset of randomly selected local datasets. Only the top-performing models are included in the global ensemble. This selection mechanism ensures that the global model comprises the most reliable and informative local models, enhancing its robustness against partial and noisy label data.

## B. Model Heterogeneity

Model heterogeneity pertains to the variations in network architectures and learning paradigms that exist across a wide range of devices [179], [183]. In [178], the author introduces

the concept of model heterogeneity, suggesting its potential applicability. While the methodology employed in the study was evaluated within the confines of a uniform network architecture, it is noteworthy that the proposed strategy does not necessitate the adoption of an identical model architecture across all nodes. This is particularly relevant given the diverse nature of tasks encountered in real-world applications, where local devices may not necessarily utilize the same model architecture due to varying requirements and constraints.

Additionally, to simulate and handle model heterogeneity, the authors in [170] assign different models, including ResNet, EfficientNet, MobileNet, and GoogLeNet, to each domain in the classification tasks. They then introduce the Federated Cross-Correlation Learning method, which computes the logits output from these diverse models. This approach enhances collaboration by maximizing similarity within the same dimensions of the output while minimizing the correlation between different dimensions. This approach enables effective collaboration among participants with varying model architectures using shared public data to learn a generalized representation.

In [171], the author extends the method proposed in [170] by ensuring that each participant in the federated learning environment employs a distinct model, thus creating a heterogeneous federated environment. The novel method utilizes the embedding features produced by these models to align the instance similarity distribution across participants. This facilitates communication at the feature level, thus improving collaboration while preserving the each unique model.

## C. Computation Heterogeneity

Computation heterogeneity refers to the differences in computing capabilities among devices. These differences can arise from variations in hardware specifications, processing power, memory capacity, battery life, and other device-specific constraints. In [172], the impact of heterogeneous edge devices is highlighted, specifically extending training time and reducing performance. The experiment included adding 10 CPU-based devices to a cluster with 20 Jetson devices. The CPU-based devices were Raspberry Pi devices with limited computing power, consisting of one with 2 GB memory, five with 4 GB memory, and four with 8 GB memory. In contrast, Jetson devices had strong parallel computing capabilities. The authors propose to address this challenge through its gradient integration method, which optimizes the model update process to be accurate and efficient. The method utilizes a knowledge extractor to retain critical model weights as task knowledge, reducing the computational footprint. During training, the gradient integrator combines the gradient of the current task with gradients of previously learned signature tasks, ensuring that the updated model weights do not degrade performance on previous tasks. The results showed that training on resource-constrained Raspberry Pi devices delayed the training time of all techniques by an average of 12 times. Moreover, computational heterogeneity reduced the accuracy of all tested algorithms by 3% to 5%.

Similarly, in [173], the authors study the satellite-airborne-terrestrial edge computing networks, devices such as IoT devices, unmanned aerial vehicles, and satellites, which possess

diverse computing capabilities due to their different hardware specifications and deployment environments. This heterogeneity in computing power leads to significant variations in the amount of CPU resources available for data processing. For instance, an IoT device may have a CPU frequency ranging from 315MHz to 916MHz, whereas a satellite can have CPU frequencies as high as 1GHz to 10GHz. This variance in computational capabilities poses challenges in devising efficient resource allocation policies that optimize both the age of information and energy consumption across these heterogeneous devices. To tackle this problem, a lifelong learning algorithm is used to optimize the trade-off between the age of information and energy consumption. It models the problem as a sequence of tasks where each device in the network experiences changes in its operating environment. These tasks are formulated as Markov decision processes. The lifelong learning approach allows the network to learn policies for new tasks more efficiently by reusing knowledge from previously learned tasks. A high-altitude platform acts as a knowledge base, collecting data from devices, performing lifelong learning, and transmitting updated policies back to the devices. Experimental results show that the algorithm significantly accelerates the learning process compared to traditional reinforcement learning algorithms.

#### D. Communication Heterogeneity

Communication heterogeneity refers to the differences in communication capabilities and network conditions among devices participating in NCL. These differences can arise from variations in network bandwidth, latency, connectivity stability, and data transfer protocols, impacting how efficiently devices can exchange information. Based on whether the task orders are the same among clients, communication heterogeneity can be categorized into two main types: synchronous and asynchronous communication.

1) *Synchronous*: In [174], the issue of synchronous communication arises when clients are required to proceed through the same sequence of class-incremental tasks in a lockstep manner. This synchronization can lead to inefficiencies and bottlenecks, especially when clients have varying computational capabilities and data availability. The challenge can be formalized as ensuring that all clients complete the task in the same incremental state before proceeding to the next task. This is often hampered by data heterogeneity. To address this synchronous communication issue, the authors propose to incorporate a prototypical knowledge fusion mechanism on the server side. This mechanism allows for spatial knowledge transfer among clients by constructing a global knowledge base that aggregates prototypical information from all participating clients. The fusion process horizontally aggregates heterogeneous prototypical knowledge from different clients and vertically fuses previous and new knowledge along the timeline. By leveraging prototypical knowledge fusion, the method enables clients to proceed asynchronously while still maintaining a global consensus on the learned knowledge. The server periodically distributes the updated global knowledge base to clients, ensuring that each client's local model can

incorporate the latest global prototypical information. It's a typical setting implemented in the previous works [180]. For instance, after apportioning ten classes across five tasks and assigning them to three clients, when a new task arrives, each client will acquire an additional two classes consistent across clients, yet the samples within these classes are distinct.

2) *Asynchronous*: In [175], the authors first formulate the asynchronous federated continual learning scenario; the primary challenge arises from the asynchronous nature of communication among clients. Each client follows its own task stream and learns from new classes at its own pace, leading to out-of-sync local data streams. This asynchrony can cause catastrophic forgetting, where the global model loses knowledge of previously learned classes as new classes are introduced asynchronously across clients. This is represented by the class set for each client at each round, which can differ significantly from client to client.

To tackle this asynchronous communication issue, the FedSpace framework employs a prototype aggregation mechanism. At each communication round, each client computes the prototype and radius for each class in its current task set. These prototypes are then aggregated on the server side into global prototypes, which capture common knowledge across clients. By aggregating prototypes instead of raw data, FedSpace preserves privacy while enabling knowledge transfer across asynchronously updating clients. Furthermore, FedSpace introduces a contrastive representation loss to align the old aggregated prototypes with the new locally learned representations. This loss function encourages feature vectors of the same class to be close together while pushing apart feature vectors of different classes. This helps mitigate catastrophic forgetting by ensuring the new local representations remain consistent with the global knowledge base, even as clients learn new classes asynchronously.

**Conclusion.** In this section, we analyze four common heterogeneity issues in distributed learning: data, model, computation, and communication heterogeneity. As an inevitable challenge in practical distributed learning systems, addressing heterogeneity can enhance algorithm robustness. Overall, research on heterogeneity in NCCL predominantly focuses on data heterogeneity, while studies on other heterogeneity issues remain relatively scarce, mostly confined to preliminary explorations without in-depth investigation of their unique impacts. In contrast, NCL has achieved more mature investigations across all four heterogeneity issues. Integrating these insights with CL, future research directions could prioritize: 1) Communication heterogeneity under sequential task arrival scenarios: The asynchronous arrival of new tasks across clients and the incompatibility of direct model aggregation require systematic solutions; 2) Computation heterogeneity for synchronization efficiency: Addressing varying task update speeds across clients through computation-aware methods could improve model robustness in NCCL; 3) Model heterogeneity for practical deployment: Developing algorithms compatible with diverse model architectures (driven by policy or environmental constraints) could facilitate NCCL implementation across heterogeneous devices.

## V. DISCUSSION ABOUT SECURITY AND PRIVACY

In the previous section, we focus on analyzing the performance of the NCCL method across different techniques and heterogeneity issues. Given the nature of NCCL, where learning occurs on distributed devices, it is essential to address the security and privacy issues during knowledge transmission. In this section, we explore the critical aspects of privacy and security within the framework of NCCL. Specifically, we will analyze several types of malicious attacks and the defense methods that can be employed to safeguard the integrity of local private data, which is shown in Table VII.

- *Malicious Attack:* Several malicious attacks in NCCL systems will be introduced that aim to disrupt the learning process or extract valuable information. We will detail data poisoning attacks [184], where adversaries introduce harmful data into the training set, and backdoor attacks [185], which involve embedding a hidden, manipulatable feature within the model.
- *Defense Method:* To mitigate the threats posed by privacy leakage and malicious attacks, it is essential to explore the defensive strategies implemented in NCCL. We will discuss differential privacy techniques that add controlled noise to protect individual data points [186], the use of order-preserving encryption to maintain data order after encryption [187], and the incorporation of anomaly detection [188], adversarial training [189], and shadow learning to prevent potential attacks [185].

### A. Malicious Attack

We focus on two typical attack methods (data poisoning and backdoor attack) aimed at disrupting the learning process or stealing valuable information.

1) *Data Poisoning:* It often uses an adversarial strategy where the attacker introduces malicious data into the training set of the distributed device [194]. The goal of data poisoning is to degrade the performance of the trained model or to manipulate its behavior. The authors in [184] analyze data poisoning attacks in NCCL in detail. The Gaussian noise attack, where an adversary introduces noise drawn from a zero-mean Gaussian distribution into the model parameters, is employed. The effect of this attack is to disrupt the model's ability to learn from the data, leading to increased variance in the model's predictions and a potential decrease in accuracy. Additionally, the label-flipping method is discussed, which involves the alteration of training sample labels to disrupt the learning process. The impact of label flipping is to mislead the model, causing it to learn incorrect associations between inputs and outputs, which can significantly impair the model's ability to generalize to unseen data.

2) *Backdoor Attack:* It involves inserting a backdoor into a model during training, which can later be activated to manipulate the model's behavior [190]. In [185], the authors discuss a backdoor attack in NCCL, where an adversary manipulates the training process to embed a hidden behavior into the model. One common approach is the pixel attack, where a malicious client adds a specific pattern or trigger to a subset of training samples and alters their labels to a target label. The goal is

to ensure that when the model encounters a sample with the trigger, it will classify it as other labels, regardless of the actual content of the sample. Another variant discussed is the backdoor attack with evasion of anomaly detection, where the adversary modifies their objective function to include an anomaly detection term, aiming to evade defenses.

### B. Defense Methods

Existing NCCL research has primarily focused on defense methods to ensure the security of distributed data during transmission. We have identified four relevant studies: differential privacy, encryption, anomaly detection, and shadow learning.

1) *Differential Privacy:* It is characterized by the introduction of controlled noise into data or its derivatives, which guarantees that the addition or removal of a single record from a dataset will not significantly affect the prediction [195].

Similar to [196], authors in [186] introduce a novel approach to address privacy challenges and heterogeneous privacy budgets in federated learning. The method leverages differential privacy to ensure strong privacy guarantees while allowing clients grouped into cohorts to define their own privacy requirements. To train a differentially private model, the paper adapts the DP-stochastic gradient descent algorithm to incorporate heterogeneous privacy budgets. At each communication round, the algorithm tracks the privacy loss for each cohort using a cohort-based privacy accountant. Additionally, it introduces two novel continual learning-based DP training methods, DP-synaptic intelligence and DP-rehearsal, to mitigate the effects of forgetting previously learned experiences when the privacy budget of a cohort is spent. These methods regularize the learning process, reducing the performance drop in stricter privacy cohorts.

The method proposed in [197] employs the Laplace mechanism to add noise to local model updates, ensuring that individual contributions from edge devices remain obscured. This approach not only protects user data privacy but also maintains model performance by balancing the trade-off between privacy and utility. To incorporate DP, each device adds Laplace noise to its local gradient updates. The server then aggregates these noise-added updates to form the global model. This method ensures that the global model retains the ability to adapt to new tasks while preserving previously learned knowledge, thereby effectively mitigating catastrophic forgetting. Extensive experiments on the MNIST dataset demonstrate that the proposed method achieves a favorable trade-off between privacy protection and model accuracy, particularly under non-IID data distributions.

Furthermore, [198] introduces a blockchain-based differential optimization federated incremental learning algorithm to enhance privacy preservation in NCCL. The algorithm employs differential privacy techniques, specifically the addition of Laplace noise, to safeguard sensitive data during the model training phase. The core idea is to ensure that the model parameters do not leak sensitive information, which is achieved by adding noise to the model updates. The Laplace mechanism is used to add noise to the model's output. This approach helps in reducing the impact on the model's accuracy

TABLE VII  
A SUMMARY OF SECURITY AND PRIVACY ISSUES IN NCCL.

Ref	Key Ideas	Advantages	Limitations
<b>Security and Privacy Issue: Malicious Attack</b>			
[184]	Utilizing Gaussian noise and label flipping methods to analyze data poisoning attacks in federated continual learning.	Gaussian noise reveals basic attack impact.	Label flipping struggles with task-specificity in FCL.
[190]	Investigating tailoring byzantine attacks and a novel incremental forgetting attack to disrupt federated continual learning models.	Introduces novel attacks tailored to NCCL vulnerabilities.	Effectiveness varies with defense mechanisms and settings.
[185]	Discussing backdoor attacks in federated continual learning, along with pixel attack and anomaly detection evasion approaches.	Exploits backdoor leakage to degrade model robustness.	Requires sustained malicious influence over training rounds.
<b>Security and Privacy Issue: Defense Method</b>			
[186]	Using cohort-based differential privacy to ensure varying privacy requirements and data distributions among different clients.	Introduces cohort-level DP to protect heterogeneous privacy budgets.	Relies on cohort structure and may degrade with dynamic data.
[191]	Using Bayesian differential privacy to provide more balanced privacy protection for different streaming tasks.	Dual-model structure and Bayesian differential privacy enhance accuracy and privacy.	Model compression and noise addition may introduce computational overhead.
[192]	Calculating differentially private means of local data per class for each device.	DP noise enables shared replay buffers to enhance performance.	High client participation reduces gains from the global buffer.
[187]	Utilizing a practical order-preserving encryption mechanism designed for vertical federated continual learning.	Efficient privacy protection with OPE maintains model accuracy.	Potential distribution leakage, overheads with sparse data.
[193]	Introducing a GRU-based anomaly detection framework to counteract data poisoning attacks in IoV environments.	GRU-based anomaly detection in NCCL for dynamic IoV.	Relying on the delay feature, threshold selection is critical with specific datasets.
[185]	Employing a robust defense against backdoor attacks using data filtering, early stopping, and periodic retraining.	Uses shadow learning to filter and early-stop backdoor attacks.	May struggle with high client heterogeneity or dynamic data distributions.

due to the addition of differential privacy. The paper discusses optimizing parameters within a weighted random forest to mitigate the accuracy loss associated with differential privacy. The algorithm also incorporates an ensemble learning technique to integrate local models, improving the global model's accuracy. The model parameters are uploaded to the blockchain, ensuring secure and efficient parameter management. This method not only bolsters the model's robustness against adversarial attacks but also leverages blockchain technology to ensure data integrity and non-repudiation.

2) *Order-Preserving Encryption*: It is a cryptographic technique that allows encrypted data to be sorted or compared while maintaining the original data order [199]. The core property of OPE is that for any two values  $i, j$  in a range  $R$ , if  $i < j$ , then the encrypted values  $f(i) < f(j)$ , where  $f$  is the OPE function. This property is crucial for decision trees, which compare feature values to make splitting decisions.

In [187], order-preserving encryption (OPE) is employed to protect data privacy while enabling the training of decision trees in federated incremental learning. Traditional OPE algorithms may provide weaker security than common encryption methods, as they preserve the size relationships of the ciphertext. To mitigate this, a practically order-preserving encoding mechanism is proposed to add random noise to the encrypted data to enhance privacy. This noise helps to achieve differential privacy, which is a formal framework for protecting the privacy of individual data points in a dataset. To further improve the efficiency of the solution and reduce storage overhead, a regional counting method is introduced. This method divides the values of a feature into different groups and uses the group mean to denote the feature value

of the group. The encryption function is then applied to these group means rather than to individual data points. This strategy not only maintains model accuracy but also enhances privacy by reducing the granularity of the data that is exposed to potential attackers. The use of OPE in regional counting allows for the training of federated incremental decision trees with strong privacy guarantees while maintaining high efficiency.

3) *Anomaly Detection*: It is defined as the process of identifying clients or data points that significantly deviate from the expected behavior or model updates, which may indicate malicious activities or data corruption [200], [201].

For anomaly detection in Kubernetes environments, the holistic security and privacy framework [202] employs NCCL with unsupervised machine learning, notably auto-encoders, to detect anomalies directly from raw network traffic, sidestepping the need for labeled data and preserving privacy. This framework comprises three main components: the collector, which gathers network traffic data; the agent, responsible for local analysis and model training; and the aggregator, acting as the central server to coordinate federated training and model updates. This architecture enables the framework to process and classify the network traffic in real-time, leveraging unsupervised learning to refine models based on unlabeled data and to adapt to new patterns without forgetting previously learned knowledge. The evaluation demonstrates its effectiveness in a micro-services-oriented application environment, where it was tested against various types of attacks, including denial of service, port scan, brute force, and SQL injection [203].

To detect accounting anomalies in financial auditing, [204] is designed to address the challenge of learning highly adaptive audit models in decentralized and dynamic settings, where

data distribution shifts over multiple clients and periods are common. The anomaly detection is performed using auto-encoder networks, which are trained to learn a comprehensive model of the data distribution. This architecture consists of an encoder and a decoder. The encoder learns a representation of a given input sample, and the decoder learns a reconstruction of the original input. The model is optimized to minimize the reconstruction error. In the audit setting, the reconstruction error magnitude is interpreted as the deviation from regular posting patterns, and journal entries with high reconstruction errors are selected for detailed procedures.

4) *Shadow Learning*: It provides a robust defense against backdoor attacks by integrating data filtering, early stopping, and periodic retraining of the shadow model [205]. Two models will be trained in parallel: a backbone model is trained on all client data and aims to achieve high performance on the main task without specific defense mechanisms while a shadow model is central to the defense approach. The authors in [185] further optimize shadow learning for federated continual learning. The backbone model undergoes continual training with data contributions from all clients, unburdened by any defensive mechanisms, thereby focusing solely on executing the primary classification task. However, given the omnipresent risk of backdoor attacks, where adversarial clients may introduce malicious training data, the backbone model may gradually become compromised. To counteract this peril, shadow Learning introduces a filtering mechanism rooted in robust covariance estimation and data whitening [206].

The filtering process hinges on computing the quantum entropy score of sample representations, a metric derived from the spectral properties of the data. This score serves as a discriminant tool, amplifying the spectral signature of corrupted data, thus facilitating the identification and removal of malicious updates. After the filtering process, the resultant clean updates are employed to nurture the shadow model. This model undergoes training until an early stopping criterion is met, thereby averting overfitting to potential backdoor triggers. The shadow model, thus, functions as a pristine version of the backbone model, specifically tasked with detecting and accurately classifying targets potentially tainted by backdoor attacks. During the inference phase, input samples are initially processed by the backbone model. If the predicted label falls within a predefined set of suspicious target labels (determined through the filtering phase), the sample is subsequently passed through the shadow model for definitive classification. This dual-model approach ensures the system maintains high accuracy on the primary task while effectively fortifying its defenses against backdoor attacks.

**Conclusion.** This section examines security and privacy concerns through the lens of malicious attacks and defense methods, along with their associated techniques. Current research predominantly focuses on defense strategies, while investigations into attack methodologies remain limited, primarily combining conventional NCL attack approaches with continual learning scenarios. Future studies should prioritize exploring unique challenges inherent to the NCCL paradigm, such as inter-task correlations within streaming tasks and their impli-

cations for designing attack-resilient defense algorithms.

## VI. DISCUSSION ABOUT REAL-WORLD APPLICATIONS

In the preceding sections of this survey, we have discussed the foundational issues and methodologies of NCCL in enabling collaborative continual learning among distributed devices. This emerging paradigm enables distributed systems to adapt continuously to new data while preserving previously acquired knowledge with privacy preservation. In this section, we turn our focus to practical applications of how NCCL can be effectively utilized in the real world to solve downstream tasks and cope with resource constraints.

### A. Downstream Tasks

The growing complexity and diversity of tasks in real-world applications [207] highlight the importance of NCCL. Despite extensive academic research conducted on NCCL, practical applications require greater attention to challenges arising from privacy concerns, communication overhead, and scalability issues, especially amidst rapid data growth and real-time demands from various devices. This section focuses on four key application areas where NCCL can have a substantial impact, summarized in Table VIII and IX.

- *Internet of Things (IoT)*: In the era of pervasive computing, the Internet of Things has become a critical infrastructure, with countless IoT devices continuously generating and processing data, leading to an explosive growth of distributed data [208], [209]. NCCL enables IoT devices to continuously learn and adapt to new tasks or environments in a collaborative manner without the need for data centralization, thereby greatly enhancing the intelligence and autonomy of IoT systems while reducing privacy risks and communication costs [210], [211]. We depict the application of IoT devices in Fig. 9 in terms of hierarchical learning with the client-edge-cloud model.
- *Intelligent Transportation Systems (ITS)*: Modern transportation systems generate massive amounts of data from various sources, such as vehicles, roadside units, and traffic cameras [212]. The unique challenge lies in the real-time processing system while ensuring data privacy and system scalability. NCCL can be utilized to develop intelligent transportation systems that continuously learn and adapt to changing traffic conditions, user behaviors, and environmental factors while keeping sensitive data local [213], [214]. This can lead to more efficient and sustainable ITS. Fig. 10 depicts the scenario of transportation systems with federated learning.
- *Medical Diagnosis*: Medical diagnosis technologies, such as X-ray, CT, and MRI, generate a substantial volume of critical data in disease diagnosis and treatment [215], [216], requiring careful handling to protect patient privacy and data integrity. NCCL enables distributed medical institutions to collaboratively learn and improve diagnostic models without sharing raw patient data [217], [218]. This can accelerate the development of more accurate and robust medical imaging AI while safeguarding patient privacy. Fig. 11 illustrates the application scenario of

medical images in terms of decentralized learning without a central server.

- *Network Application:* Communication technologies such as 5G/6G open radio access networks and digital twin infrastructures generate vast amounts of critical data for network management and anomaly detection. NCCL enables distributed network operators to collaboratively learn and optimize diagnostic and anomaly detection models without sharing raw data [219], [220]. This facilitates the rapid development of more accurate and robust artificial intelligence systems for both 5G network anomaly detection and digital twin applications while simultaneously safeguarding data integrity and privacy.

1) *IoT:* In [221], authors propose a federated incremental learning method for data sharing in the industrial IoT. This addresses the common challenges of high new data influx and volume imbalance among factory nodes. This approach integrates new state data seamlessly with existing industry federation models. It begins by distributing an initial local model to plant sub-terminals. To handle dynamic and uneven data distribution, a federated sub-terminal optimization algorithm adjusts participating nodes for fair and efficient training. The core of this is computing incremental weights via a federated incremental learning algorithm. This allows rapid data integration, enhancing system adaptability and responsiveness. The method is tested on the CWRU bearing dataset [236], outperforming traditional FedAvg and non-incremental methods in diagnostic accuracy.

Similar to [119], the authors in [237] use a federated incremental learning framework combined with knowledge distillation to address the modulation classification challenge in the context of cognitive IoT, aiming to effectively integrate the classification knowledge from the private categories of local devices into the global model. This method regards the private classes of a particular device, which are different from the common classes among all devices, as incremental classes. Incremental learning is adopted during the training of each local model to learn the classification knowledge of these private classes. Moreover, knowledge distillation is utilized in the local model training stage to maintain the classification knowledge from the global model and avoid excessive dispersion between the global model parameters and local model parameters.

In both [119] and [237], knowledge distillation is leveraged to adapt tasks effectively. The first study employs models from both the previous client round and the current server as teacher models, encapsulating the necessary knowledge for assessing past tasks. By calculating the distillation loss between these teacher models and the current client’s model, the method retains prior knowledge. The second study treats each client’s unique class data as incremental, using knowledge distillation during local training to preserve global model classification knowledge. Here, local models are trained to excel on their tasks and mimic the global model’s behavior. This dual objective minimizes a combined loss function of local data loss and distillation loss. Both methods integrate classification and distillation losses to mitigate forgetting.

In real-world IoT traffic classification, the authors in [238]

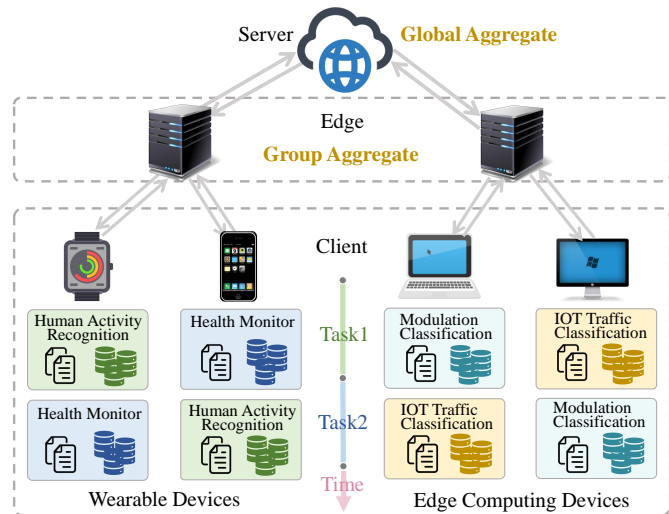


Fig. 9. Hierarchical learning scenario with continual learning of IoT devices scenario. The IoT devices are divided into multiple groups to continuously learn new tasks, and each group will rely on the edge aggregation model at the side end and, ultimately, on the server aggregation on the cloud.

introduced Fed-SOINN, a novel attention-based federated incremental learning algorithm. By integrating incremental and federated learning, Fed-SOINN collaboratively trains a comprehensive network classification model while preserving data privacy in non-IID IoT environments. Based on the SOINN algorithm and radial basis function, Fed-SOINN’s online traffic classification model adapts to time-varying data. This semi-supervised approach identifies unknown traffic categories and utilizes an attention mechanism to enhance the weights of beneficial client model parameters. By sharing only important gradients, Fed-SOINN reduces communication rounds. Evaluated on Moore [239] and ISCX-VPN datasets [240], Fed-SOINN improved detection accuracy by 3.1% and reduced communication rounds by up to 73%. Additionally, its incremental learning mechanism effectively identifies new traffic categories, ensuring stable overall performance.

In [241], the authors aim to address data freshness in IoT and mobile edge computing. It prioritizes newer data batches through a loss compensation mechanism, ensuring they have a more substantial influence on model updates. Suitable for continuous data flows from IoT devices, the method adjusts the loss function based on data batch arrival times, enhancing the model’s responsiveness to recent data pattern changes. Implementing a batch-oriented aggregation method updates federated models more frequently and at a finer granularity, aligning with the dynamic nature of IoT data. Compared to traditional approaches, the method significantly improves convergence speed and learning performance.

2) *Intelligent Transportation Systems:* In [233], the authors propose ICMFed, an incremental and cost-efficient federated meta-learning mechanism for real-world 3D driver distraction detection in transportation systems. ICMFed tackles four key challenges: data accumulation, communication optimization, data heterogeneity, and device heterogeneity. By integrating incremental learning, meta-learning, and federated learning,

TABLE VIII  
NCCL FOR REAL-WORLD APPLICATIONS (1/2).

Ref	Key Ideas	Advantages	Limitations
<b>Application: Internet of Things (IoT)</b>			
[221]	The factory participates in the federated sub-terminal selection with dynamic local data.	Data sharing in a privacy-preserving manner.	Complexity in parameter depth management.
[222]	Enhancing robot navigation within cloud robotics through the application of knowledge transfer within federated continual learning.	Efficient knowledge fusion with different robots.	High computational overhead.
[223]	Implementing federated learning with continuous sim-to-real transfer for vision-based obstacle avoidance in IoT robotic systems.	Enhanced model generalization in obstacle avoidance.	Mismatching between simulation and reality.
[224]	Utilizing class gradient balance and label smoothing loss in conjunction with federated continual learning for dynamic IoT intrusion detection.	Dynamic memory and balanced loss design.	Limited non-IID adaptability.
[225]	Employing a discriminative auto-encoder and federated continual learning for incremental intrusion detection in IoT networks.	Reduced communication overhead.	Dependence on partial historical data.
[202]	Achieving decentralized anomaly detection through continual updates leveraging unsupervised machine learning.	Efficient continual model adaptation.	Inconsistent performance across different scenarios.
[226]	Enhancing decentralized learning with iterative model aggregation for continuous deepfake video detection in the entertainment industry.	Resilience against single points of failure.	Slightly lower accuracy with increased devices.
[227]	Utilizing satellites as agents in a deep reinforcement learning framework, with federated continual learning enabling adaptive routing in satellite networks.	Adaptability to dynamic network environments.	Complexity in model synchronization across satellites.
[228]	SurveilNet leverages federated learning for collaborative anomaly detection in surveillance systems with streaming data while ensuring data privacy.	Efficient privacy preservation.	Limited scalability under large-scale deployments.
[229]	A failure-aware framework for drone delivery incorporates weighted FL for predictive maintenance and a speed-heuristic approach for real-time service.	Enhanced failure prediction with FL.	Increased computational overhead in large-scale deployments.
[230]	Addressing concept drift and privacy in photovoltaic power forecasting through energy-distance-aided drift detection and incremental federated broad learning.	Efficient concept drift detection.	Potential high communication overhead.
[231]	Enhancing fault prediction in power networks through personalized federated learning and incremental SCNs, dynamically integrating data across substations.	Incremental SCN for dynamic streaming data.	Sensitivity to extreme data heterogeneity.
[232]	The blockchain-enhanced algorithm ensures the security of digital twin networks through decentralized learning and robust privacy protection.	Blockchain-enhanced data security and privacy.	High communication and storage overhead.
[204]	Continual learning facilitates the detection of accounting anomalies in financial audits by collaboratively updating models across decentralized clients.	Privacy-preserving federated continual learning framework.	Fail with internal adversarial attacks.
<b>Application: Intelligent Transportation System (ITS)</b>			
[233]	Federated incremental learning integrated with meta-learning and batch-dependent temporal factors for advanced driver distraction detection in 3D space.	Efficient communication cost reduction via gradient filtering.	High computational overhead on client devices.
[234]	Point-to-point federated continual learning for accurate traffic condition prediction.	Enhanced privacy protection through decentralized data processing.	Limited model generalization under heterogeneous sensor data.
[235]	A vehicle diversity selection algorithm that leverages multiple performance indicators for enhanced intelligent connected vehicles systems.	Improved performance while preserving privacy.	Dependence on stable network coordination.
[213]	A fog-cloud system in FL boosts the accuracy of ITS event detection by processing local vehicle data and sharing global insights among clients.	Efficient communication via the fog-cloud framework.	Potential communication overhead between fog and cloud.
[214]	Connected vehicles engage in consensus-driven federated learning to collaboratively refine deep neural networks collaboratively, enabling extended sensing capabilities in 6G environments.	Communication-efficient modular design.	High Dependence on connectivity density.

ICMFed introduces a novel learning paradigm for 3D tasks. It stabilizes local model training with a temporal factor associated with local batches and retains accumulated knowledge. A gradient filter optimizes client-server interaction and reduces communication costs. Additionally, a normalized weight vector enhances global model aggregation, considering the richness of local updates. ICMFed supports rapid user personalization by adapting the global meta-model to specific contexts. The framework was validated using the State-Farm-Distracted-Driver-Detection dataset, which contains 22,424 labeled samples from 26 drivers. ICMFed addresses real-world

challenges in detecting unsafe behaviors like unfocused eyesight and inattention, achieving 96.86% quality improvement while protecting user privacy through federated updates.

In [234], the authors introduce federated point-to-point continual learning, a federated peer-to-peer strategy for continuously learning from traffic intensity sensor data in urban areas. It aims to collaboratively develop a robust model capable of predicting traffic conditions in non-static urban environments. Leveraging edge computing, this paradigm processes data locally on IoT devices like traffic sensors. Its peer-to-peer architecture distributes the learning process across devices, fostering incremental model growth through collective

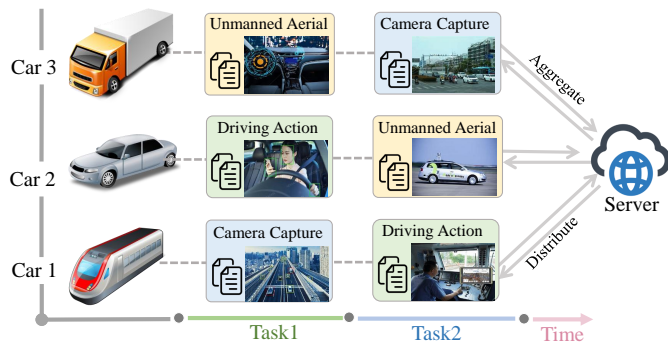


Fig. 10. Federated learning for transportation systems with the central server. Each transportation system end client needs to send the model to the server for aggregation after learning the task flow.

contributions. As new sensor data is acquired, this method dynamically updates the traffic model, ensuring accuracy and relevance without needing full retraining. The framework was tested on real traffic data from Madrid's Salamanca district, processing 127 sensors' 15-minute granularity traffic flow measurements. The method successfully predicted short-term traffic patterns while reducing energy consumption by 87%, demonstrating practical value for dynamic urban mobility management.

The authors in [235] proposed a federated learning framework based on incremental weighting and diversity selection to address the challenges of isolated data islands, incremental data, and data diversity. This approach integrates incremental learning and federated learning to provide a novel learning paradigm. The method introduces a vehicle diversity selection algorithm that uses various performance indicators, such as accuracy and loss, to calculate diversity scores. The algorithm effectively reduces homogeneous computing and maintains independent complementary data diversity by employing an improved cosine distance measure with a penalty factor. Moreover, this method proposes a vehicle federated incremental learning algorithm that uses an improved arc tangent curve as the decay function to realize the rapid fusion of incremental data with existing machine learning models. The incremental parameter depth value is calculated based on the proportion of incremental data and an adjustment variable. Applied to IoV scenarios with 353 million smart vehicles generating 4000 GB/day, the framework demonstrated 32% accuracy improvement on vehicle image datasets while handling heterogeneous road conditions (urban vs. suburban, weather variations). This enables real-time adaptation for driving assistance systems without centralized data collection.

3) *Medical Diagnosis*: In [242], the ADFLL framework is introduced, combining asynchronous learning, decentralized communication, and continual learning principles to optimize landmark localization in medical imaging. This approach addresses challenges faced by traditional federated learning systems, particularly with complex, privacy-sensitive medical data. Each agent in the ADFLL framework utilizes experience replay buffers to store tuples generated during training, enabling them to learn from past experiences and mitigate catastrophic forgetting through selective experience replay.

The asynchronous nature of ADFLL allows agents to train independently as soon as new data or buffers are available, which is particularly beneficial in medical settings with varying data availability. Collaboration among agents is facilitated through shared buffers, enriching the collective knowledge base while maintaining data privacy. The framework was evaluated on the BraTS 2017 dataset, involving 285 patients with brain tumors across four MRI sequences (T1, T1ce, T2, FLAIR) and three orientations. Agents trained asynchronously on heterogeneous hardware achieved a mean localization error of 7.81 mm, outperforming traditional models in landmark localization across various imaging sequences.

In [250], the authors adopted a decentralized learning model for the study of brain metastasis identification, in which no central server is involved and the data is retained in the local center to comply with privacy regulations. Each center sequentially trains the model using locally available data and then passes the model to the next center. The learning process is facilitated through a technique called Synaptic Intelligence (SI), which plays a pivotal role in preserving essential model weights across different training centers. This method facilitates effective collaboration between multiple centers without sharing the actual patient data, thus respecting privacy concerns. The mathematical foundation of SI is based on the concept that the parameters that undergo significant changes during learning phases should be considered more important and, thus, less prone to modification in subsequent learning sessions. Once the importance weights are established, they are used to penalize changes to critical parameters in subsequent training sessions, thus ensuring that the knowledge acquired from previous tasks is not forgotten. A collaborative scenario across seven medical centers was simulated for 920 cases of enhanced T1 MRI brain transfer data [251]. Using a decentralized joint learning framework, the model flows between centers via cyclic weight shifting and incorporates synaptic intelligence (SI) algorithms to prevent catastrophic forgetting. Iterative training further improved sensitivity to 0.914, matching centralized data pooling performance while reducing false positives to  $<1$  per volume.

In [243], the authors propose a novel peer-to-peer federated continual learning strategy called icP2P-FL to address improving the imaging performance of low-dose CT from multiple institutions. This approach integrates continual learning with a peer-to-peer federated learning framework to enable collaborative training of a global CT denoising model while preserving data privacy. The icP2P-FL method adopts a cyclic task-incremental continual learning strategy to ensure the model's performance across multiple institutions. In each cycle, the model has continually trained one institution after another via model transferring and inter-institutional parameter sharing. Furthermore, an intermediate controller is developed to make the overall training more flexible. It consists of a performance assessment module and an online determination module. The first module evaluates the denoising performance using quantitative measures, which are then fed into the second one to adjust the inter-institutional training order and determine the number of training rounds in each institution. The model was evaluated on the AAPM Grand Challenge dataset [252]

TABLE IX  
NCCL FOR REAL-WORLD APPLICATIONS (2/2).

Ref	Key Ideas	Advantages	Limitations
<b>Application: Medical Diagnosis</b>			
[242]	A replay technique leveraging shared experience replay buffers is employed to refine landmark localization in medical imagery.	Robust decentralized architecture.	High on-device computational load.
[180]	A Forgetting-balanced learning model addresses the challenge of federated incremental semantic segmentation in the context of medical images.	Enhanced performance while privacy-preserving.	High computational overhead.
[243]	A peer-to-peer federated continual learning strategy is introduced to enhance the imaging quality of low-dose CT scans.	Decentralized privacy preservation.	Complex communication overhead.
[244]	Federated continual learning, integrated with the Swin Transformer, is utilized for diagnosing plant leaf diseases.	High accuracy with robust data augmentation.	Limited real-world generalization.
[245]	A decentralized continual learning framework, leveraging GAN-based synthetic data, is employed for multi-center ECG studies to evaluate arrhythmia detection methods.	Privacy-preserving fake data evaluation.	High computational cost for GAN training.
[246]	MetaCL harnesses federated continual learning and blockchain technology for secure and adaptable physiological signal classification in the IoT.	Ensuring privacy in federated blockchain integration.	Residual blockchain poses privacy risks.
[217]	Incremental exemplar-based learning is proposed to manage real-time distributed medical data in computer-aided diagnosis for smart healthcare applications.	Efficient resource in federated-exemplar fusion.	Scalability challenges in exemplar management.
<b>Application: Network Optimization</b>			
[247]	Using the reservoir sampling buffer to enhance network anomaly detection in 5G Open-RAN by preserving previously learned knowledge.	Replay buffer for knowledge retention	Communication overhead
[248]	Combining federated learning with a replay memory-based continual learning strategy to enhance collaborative network anomaly detection in 5G Open-RAN.	Efficient replay memory strategy	Resource limitation
[249]	A federated continual learning framework integrated with digital twin technology to enhance bearing fault diagnosis by addressing abnormal sensor data and maintaining data privacy across distributed factories.	Enhanced fault diagnosis accuracy	High computational and memory demands
[219]	A federated continual learning framework with group signature authentication to ensure secure, efficient predictive maintenance for digital twins.	Efficient authentication with group signature	Complexity and computational overhead
[220]	Integrating the stacked broad learning system into a federated continual learning framework to facilitate continuous intrusion detection model training in unmanned aerial vehicle networks.	Efficient learning while privacy-preserving	Complexity in device selection

and three medical institutional datasets using the icP2P-FL framework for cross-protocol denoising. Dynamically adjust the training order and rounds through an intermediate controller. The denoising results on 100 cases of real low-dose CT data show that the structural similarity error is reduced to 3.2% that meets the clinical diagnosis requirements.

4) *Network Application*: In modern 5G open Radio Access Networks (RAN), anomaly detection is crucial due to increasing cyber threats such as distributed denial of service attacks. Traditional machine learning techniques, especially FL, face significant challenges, with one key issue being catastrophic forgetting, which is exacerbated in 5G networks owing to their disaggregated architecture and heterogeneous traffic. To address this problem, the authors in [247] integrate the concept of replay buffers into FL, effectively mitigating catastrophic forgetting while also ensuring the privacy of data. Within this framework, anomaly detectors are deployed as applications in the near real-time RAN intelligent controller, and the FL aggregation server is located in the non-real-time RAN intelligent controller. This setup enables distributed anomaly detection across the network while maintaining local data privacy. The replay buffer stores a fixed number of past data samples using a method called reservoir sampling. This ensures that a representative dataset is available for continual

learning, allowing the model to retain important information from prior attack patterns while adapting to new traffic types. As a result, the proposed solution improves the accuracy and stability of anomaly detection systems within RAN, addressing the evolving nature of cyber threats in 5G environments. Additionally, it supports scalable, privacy-preserving security measures that are essential for safeguarding the dynamic and rapidly evolving 5G networks.

In traditional Internet of Things scenarios that employ digital twins, sensor failures can lead to significant data loss, disrupting continuous data streams. Conventional diagnostic models, on the other hand, may struggle with erroneous sensor readings and often pose risks of privacy breaches. To address these challenges, the authors in [249] suggest an innovative solution that combines the concept of digital twins with retrieval-augmented generation-assisted large language models and federated continual learning. In the proposed framework, each factory deploys a digital twin to model the physical behavior of machine components, such as bearings, allowing it to refine and correct incoming sensor data to mitigate anomalies. When sensor malfunctions occur, the retrieval-augmented generation-assisted large language model generates virtual datasets to restore data continuity. Subsequently, local fault diagnosis models are trained on this enhanced data

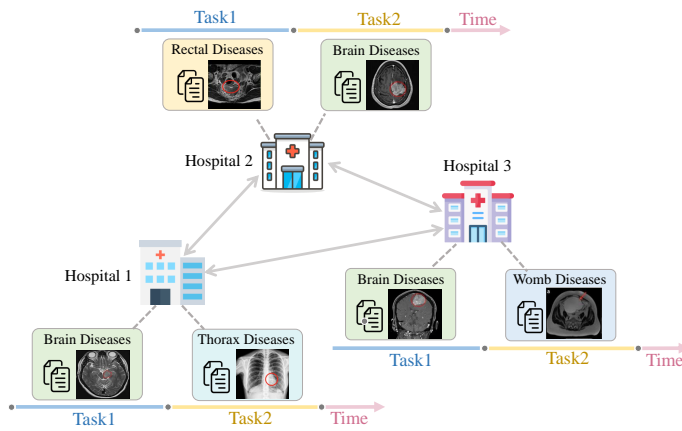


Fig. 11. Decentralized learning for multiple healthcare providers to jointly learn from streaming medical image analysis tasks, where each client exchanges learned knowledge peer-to-peer without the need for a centralized server to help aggregate the model.

using an adaptive loss function, and periodic updates to these models are aggregated through a federated continual learning process, which employs weighted averaging. This dynamic aggregation approach ensures the protection of data privacy across distributed facilities while enabling both historical and new data to effectively contribute to the global model.

In addition to bearing fault diagnosis tasks, which require knowledge retention on time-series data, the application of digital twin technology in intrusion detection systems for Unmanned Aerial Vehicle (UAV) networks also faces numerous challenges, including fragmented and heterogeneous data sources, limited computational resources, and the phenomenon of catastrophic forgetting. To overcome these issues, the authors of [220] introduce a federated continual learning framework that integrates an incremental learning approach based on a stacked broad learning system with a digital twin network. Each UAV locally collects network intrusion data and incrementally trains an intrusion detection model using the stacked broad learning system, an architecture that efficiently updates model parameters by stacking multiple learning blocks, thereby avoiding complete retraining with new data. Moreover, to ensure that only UAVs with high-quality data and adequate computing power participate in the global learning process, an intelligent selection strategy is implemented using a deep deterministic policy gradient algorithm. Selected UAVs then transmit their locally updated model parameters to a central server, where the global model is asynchronously aggregated, ensuring data privacy and robust adaptation to dynamic network conditions.

### B. Resource Constraint

In the previous section, we have already discussed the applications of NCCL in three classic real-world scenarios. While NCCL has shown promising results in enabling collaborative learning among distributed devices without compromising data privacy, its practical deployment is often hindered by the limited availability of memory, computational resources, and access to public datasets. These constraints are particularly

prevalent in edge devices, such as smartphones and IoTs, which typically have restricted storage capacity, processing power, and bandwidth [258]. Moreover, the increasing complexity and heterogeneity of real-world tasks further exacerbate the resource demands on NCCL systems. Thus, it is crucial to develop resource-efficient NCCL methods that can effectively operate under these constraints while maintaining high performance and adaptability. We consider the following three key factors to explore the resource limitations of NCCL, and Table X shows a summary of these resource-constrained scenarios of existing NCCL research.

- *Memory Buffer:* In NCCL, memory buffers are an important component for storing and retrieving previously learned knowledge [46], [45]. Existing methods usually rely on data rehearsal strategies to prevent the degradation of past knowledge, including caching samples of previous tasks or generating synthetic samples for replay so that the model can adapt to new tasks without catastrophic forgetting. However, in many real-world scenarios, such as IoT devices or mobile phones, buffers are usually limited [259]. Therefore, it is crucial to develop rehearsal-free NCCL methods that can effectively retain past knowledge while minimizing the dependency on buffers.
- *Computational Budget:* The computational budget is another critical factor in NCCL. As the number and complexity of tasks increase, the computational cost of NCCL can quickly become prohibitive, particularly for edge devices with limited processing power. This can lead to slow convergence, high energy consumption, and reduced responsiveness of the NCCL system [254]. To address this issue, it is necessary to design computing-efficient NCCL algorithms that can achieve good performance with minimal computational overhead.
- *Public Dataset:* Access to large-scale, clean, and labeled datasets is crucial for the effective training and evaluation of NCCL methods. However, in many real-world applications, such datasets may not be readily available due to privacy concerns, data scarcity, or the cost of annotation [260]. This can pose significant challenges for NCCL methods that rely on supervised learning directly from local datasets [41]. To overcome this limitation, it is important to explore an unsupervised paradigm or improve the model's robustness against noisy data that can effectively promote real-world deployment.
- *Device Deployment:* Deploying models on resource-constrained devices presents a significant challenge in NCCL. With the increasing number of devices, managing communication overhead and ensuring efficient model updates across all participants becomes complex [256]. Additionally, small devices with limited processing power and memory exacerbate the problem, making it difficult to deploy large-scale models. To address these challenges, it is essential to develop lightweight models that can adapt to resource constraints while maintaining performance [257]. Moreover, optimizing communication strategies and reducing computational requirements through techniques such as model pruning will be crucial for suc-

TABLE X  
RESOURCE CONSTRAINTS IN REAL-WORLD APPLICATIONS OF NCCL

Ref	Key Ideas	Advantages	Limitations
<b>Resource: Memory Buffer</b>			
[46]	Integrating global and local information to quantify sample importance for efficient replay.	Private modular design and efficient sample caching.	Parameter sensitivity and Static client assumptions.
[45]	Calculating both domain-representative score and cross-client collaborative score for sample selection.	Secure synergistic caching mechanism.	High computational overhead and hyperparameter sensitivity.
[180]	Combining class-aware gradient compensation loss and class-semantic relationship distillation loss with limited buffers.	Privacy-preserving prototype gradient communication.	Computational overhead from proxy server operations.
[49]	Combining category-balanced gradient adaptive compensation loss and category gradient-induced semantic distillation loss with an improved proxy server.	Adaptive gradient compensation loss.	Security reliance on proxy trust.
[253]	Combining auxiliary client-based synthetic data generation and multi-teacher knowledge distillation to mitigate catastrophic forgetting without local storage of old data.	Reduced dependence on memory buffer.	Increased computational and communication overhead due to synthetic data generation.
<b>Resource: Computational Budget</b>			
[154]	Implementing broad learning into the federated continual learning and updating the local model by the number of tasks and cached samples.	Local training for fast updates and batch-asynchronous aggregation for efficiency.	Communication overhead in batch processing and limited scalability for large client networks.
[254]	Exploring the regularization technique and improving the synaptic intelligence with balanced global and local knowledge.	Low computational overhead.	Dependence on SI algorithm.
[177]	Using an iteration-based learning rate decay strategy and gossip protocol to efficiently learn new tasks.	Low communication or computational overhead.	Resource dependence on client devices.
[255]	Dynamically adjusting parameter importance via attention-weighted aggregation of cross-task/client models with balanced global-local knowledge.	Balanced global-local learning and parameter-efficient.	Hyperparameter sensitivity.
<b>Resource: Public Dataset</b>			
[44]	Utilizing the normalizing flow model to construct the distribution and avoid introducing noisy data information.	Selective forgetting via probabilistic correlation estimation.	Computational overhead from generative model training.
[41]	Using public datasets as surrogate datasets to perform knowledge distillation at the client and server levels respectively.	Client division mechanism for balanced learning.	Dependence on public surrogate dataset quality.
<b>Resource: Device Deployment</b>			
[172]	Integrated signature task knowledge and gradient optimization to preserve task-specific information while minimizing negative knowledge transfer from non-IID data.	Lightweight and scalable client-side solution.	Dependence on hyperparameter tuning.
[256]	Leveraging sample-based signature task knowledge and gradient optimization, enabling lightweight, accurate ViT training on edge devices while minimizing communication overhead.	Lightweight client-side solution with sample-based knowledge extraction.	Impact of the hyperparameter selection.
[255]	Using an energy-efficient federated continual learning framework to optimize model training on edge devices with minimal energy usage.	Energy-efficient and suitable for small devices.	Limited scalability.
[257]	Mitigating catastrophic forgetting and system overhead by utilizing expandable models and dynamic sparse training.	Reduced system overhead and suitable for small devices.	Model complexity in large tasks.

successful deployment in real-world settings.

1) *Memory Buffer*: In [46], the authors considered the data heterogeneity and the limited storage space of the client memory buffer and proposed a simple and effective framework to solve the catastrophic forgetting with synergistic replay. The core idea of Re-Fed is to coordinate clients to cache important samples of previous tasks for replay under the limited memory of the client before the arrival of the new task. Specifically, when a new task arrives at the client, samples from previous tasks are cached according to their global and local importance. Then, the client trains the local model using both the cached samples and the samples from the new task. To quantify the importance of samples, the authors introduce an additional personalized informative model that integrates knowledge from the global model and the local model. Then, after the client receives the global model, the PIM is updated

iteratively using a momentum-based method. In the process of updating the model PIM, the gradient norm of each sample is recorded to calculate the sample importance score. Based on the importance scores, each client caches the most informative samples within its memory buffer limit. The cached samples are then used for replay during the training of new tasks to alleviate catastrophic forgetting.

Similar to [180], the authors in [49] proposed a novel Local-Global Anti-forgetting (LGA) model. This method is also designed to solve the problem of alleviating the catastrophic forgetting of the global model on old categories when the local client has limited memory. Compared with GLFC, both consider the class imbalance of new and old categories of the local client and the imbalance of non-IID categories between clients, and both use a proxy server to select the best old global model for the local client to perform knowledge distillation to solve

the global forgetting caused by the imbalance of non-IID categories across clients. The difference is that LGA introduces a category-balanced gradient adaptive compensation loss and a category gradient-induced semantic distillation loss. For global forgetting, LGA improves the proxy server by introducing a self-supervised prototype augmentation strategy to augment the reconstructed perturbed prototype images, thus enhancing the robustness in selecting the best old global model, while GLFC does not include this enhancement strategy. Extensive experiments on representative datasets demonstrate that LGA outperforms state-of-the-art methods by 4.8%~27.3% in terms of average accuracy. Moreover, the ablation studies validate the effectiveness of each module in the proposed model, especially the category-balanced gradient-adaptive compensation loss and category gradient-induced semantic distillation loss.

2) *Computational Budget*: In [154], the authors introduced a federated continual learning scheme with a broad learning framework that effectively addresses computational constraints in real-world scenarios. Central to their approach is the utilization of the broad learning paradigm, which enables efficient incremental learning without requiring full retraining of local models by each client upon encountering new data, thereby reducing computational overhead. This framework operates on the principle that local training initiates when the influx of new samples exceeds a predefined threshold. During local updates, the update of local weights for client  $i$  in update  $j$  is computed using a specific equation that adjusts the prior weights  $w$  based on the difference between new labels and model predictions, incorporating updated feature and enhancement nodes of the incremental network.

In [177], the authors proposed a novel peer-to-peer federated learning framework to address the resource constraints and privacy protection issues in driver behavior recognition tasks for the internet of vehicles. From the perspective of computational budget, the method employs a continual learning paradigm to propagate knowledge directly among clients, avoiding the computational overhead caused by central server aggregation. Specifically, when a new task arrives, each client first receives the model parameters from the previous client as initialization and then fine-tunes the model on its local data. To balance personalization and generalization, it introduces a loss function that consists of two parts: a negative log-likelihood loss and a proximal term. Moreover, the authors adopt an iteration-based learning rate decay strategy to accelerate convergence and prevent catastrophic forgetting. In terms of communication, the method utilizes a gossip protocol to randomly propagate models among clients, overcoming the highly dynamic nature and distance limitations of vehicle connections. Experiments on two real-world driver behavior recognition datasets (State-Farm [181] and AICity [261]) demonstrate that FedPC is highly competitive compared to traditional FL in terms of performance, knowledge dissemination rate, and compatibility with new clients.

3) *Public Dataset*: In [41], the authors introduce a novel continual federated learning framework known as CFED, which ingeniously leverages public datasets as surrogate data to perform knowledge distillation both at the client and server levels, thus effectively strengthening the local datasets.

Each client and the server maintain an independent surrogate dataset, which is not directly derived from the client's private data, thus addressing privacy concerns effectively. When introduced to a new task, each client utilizes its surrogate dataset to transfer knowledge from the previous tasks' model to the new model, thereby alleviating inter-task forgetting.

To mitigate the impact of noisy data in streaming tasks, [44] proposes a novel method called accurate forgetting federated continual learning. The core of this method lies in the application of a normalizing flow model to quantify the credibility of previous knowledge and achieve accurate forgetting. The model is trained in the feature space of the classifier to generate features that represent the previous tasks. These generated features are then used to augment the learning process of the classifier, preventing complete forgetting of previous tasks. The correlation probability of the generated features is used to re-weight the loss objective, enabling the classifier to selectively forget biased or irrelevant features. Specifically, the loss objective is composed of three terms: the classification loss on real data, the classification loss on generated features, and a regularization term that penalizes large deviations of the generated features from the true data distribution. The weights of these terms are adjusted based on the correlation probability, allowing the classifier to adaptively forget erroneous information and learn unbiased features.

4) *Device Deployment*: In [172], the authors proposed a client-side federated continual learning framework from the perspective of handling device number expansion. To handle the increase in device numbers, the method extracts and integrates signature task knowledge. Each client has a knowledge extractor that retains partial model weights as task knowledge, reducing computational load. A gradient integrator combines gradients from the current and relevant past tasks to prevent catastrophic forgetting and negative knowledge transfer.

Similarly, in [256], the FedViT framework is proposed for ViT models in federated continual learning. It also adopts a client-side strategy. FedViT extracts knowledge as a subset of training samples (e.g., 10% per task) to reduce device-side burden. Leveraging ViT's decomposability, it trains the model in stages. FedViT consists of three key components: knowledge extractor, gradient restorer, and gradient integrator. The knowledge extractor stores the knowledge of each task by retaining a subset of training samples that are well-adapted to the trained ViT model and maintains the class distribution of the task's complete samples. Extensive experiments on various datasets demonstrate that FedViT significantly improves model accuracy without increasing training time and reduces communication costs. Moreover, FedViT achieves more improvements under difficult scenarios such as large numbers of tasks or clients and training different complex ViT models.

In [255], the author introduces an energy-efficient framework for personalized federated continual learning, specifically designed to address the challenges of continual learning on edge devices with limited computational resources. To address this issue, the model is divided into two parts: one part retains old knowledge, while the other learns new tasks, minimizing computational costs. A sparse pruning strategy is adopted to preserve critical model components without

sacrificing performance. This sparse optimization approach reduces both memory and computation requirements, making it particularly well-suited for edge devices. The experimental results demonstrate that the proposed method significantly reduces energy consumption by training only part of the network. In terms of scalability, as the number of devices increases from 20 to 40, the proposed method continues to maintain high performance.

In [257], the author suggests a sparse personalized federated class-incremental learning framework to tackle key challenges in continual learning on edge devices with limited resources. The framework targets issues such as catastrophic forgetting and system overhead and proposes two main solutions: expandable class-incremental models and dynamic sparse training. To mitigate catastrophic forgetting, the framework replaces single-task models with expandable class-incremental models. This approach allows for task-specific blocks to be added as new tasks are learned, preventing the model from forgetting old knowledge while enhancing its capacity for new tasks. Additionally, dynamic sparse training customizes sparse local models for each client, reducing system overhead by pruning unnecessary parameters, thus mitigating data heterogeneity and over-parameterization without increasing computational costs. Personalized sparse masks optimize the model by adjusting to task requirements, ensuring efficient resource use and updating only critical components. This approach does not require additional data for knowledge distillation, making it ideal for edge devices with privacy concerns.

**Conclusion.** This section examines three downstream task scenarios of NCCL and four resource constraint issues in real-world deployment contexts. NCCL has been extensively investigated across numerous practical applications, demonstrating its critical role in economic development. Unfortunately, resource constraint issues remain under-investigated. Future research should prioritize lightweight NCCL algorithms and extend their applicability to broader real-world scenarios.

## VII. BENCHMARK EXPERIMENTS

In this section, we will establish benchmarks to empirically evaluate the performance of current NCCL methods. We are the first to construct such benchmarks for testing. Here, considering the diversity of NCCL scenarios, we only adopt task scenarios that combine FL and CL, also commonly referred to as federated continual learning, which is a mainstream research direction with substantial existing studies. Below, we will introduce the datasets, baselines, experimental configurations, and the analysis of experimental results <sup>1</sup>.

### A. Dataset

For benchmark datasets, we considered two of the most common CL scenarios: class-incremental learning (Class-IL)

<sup>1</sup>We will continually organize the relevant code and datasets, and all related methods will be integrated into a unified code framework to facilitate comparison and deployment by researchers. The code will be available at: <https://github.com/YichenLi-Hust/NCCL/>

and domain-incremental learning (Domain-IL). For each scenario, we utilized three datasets to conduct experiments. The specific details of the datasets are as follows:

**Class-Incremental Task:** Datasets in the Class-IL scenario progressively introduce new classes over time, starting with a subset of classes and expanding in subsequent stages. This enables models to adapt to an increasing number of classes.

- *CIFAR-10* [262]: A dataset includes 10 different object categories, each sample represented as a  $32 \times 32$ -pixel color image. It comprises 50,000 training samples and 10,000 test samples.
- *CIFAR-100* [262]: A dataset with 100 fine-grained classes, including 50,000 training images and 10,000 test images, provides a challenging benchmark for image recognition.
- *Tiny-ImageNet* [263]: A subset of ImageNet with 200 classes and 120,000 images (100,000 training, 10,000 validation, and 10,000 tests).

**Domain-Incremental Task:** Initially comprising instances from a particular domain, Domain-IL progressively incorporates novel domains over time. This setup effectively facilitates adaptation and generalization to previously unseen domains.

- *Digit-10*: A dataset encompasses ten categories of digits across four domains: MNIST [264], EMNIST [265], USPS [266], and SVHN [267]. Each dataset represents ten classes of digit images within a specific domain, such as handwriting, printed text, and others.
- *Office-31* [268]: A dataset consists of images gathered from three varying domains: Amazon, Webcam, and DSLR. It encompasses 31 categories of objects, with approximately 4,100 images belonging to each domain.
- *Office-Caltech-10* [269]: A dataset featuring images sourced from four distinct domains: Amazon, Caltech, Webcam, and DSLR. It encompasses 10 categories of objects, with approximately 2,500 images per domain.

### B. Baseline

For the baseline, we chose relevant work published in top conferences and journals (such as ICLR, NeurIPS, ICML, CVPR, ECCV, ICCV, etc.) in recent years to ensure the superiority and advancement of baselines. To ensure a fair comparison among baselines, we follow the protocols established in [4] and [46] to set up tasks: FedAvg [4], Re-Fed [46], FedCIL [42], GLFC [180], FOT [148], FedWeIT [40], CFed [270], Target [43], AF-FCL [44], MFCL [127], pFedDIL [101], SR-FDIL [45]. As these baselines have been extensively discussed in previous sections, we omit detailed descriptions here to avoid redundancy.

### C. Configuration

We assign different numbers of tasks to various datasets. For class-incremental tasks, using CIFAR-10 as an example, we set five tasks, with each task covering two classes without data overlap. For domain-incremental tasks, each domain represents one task. Dirichlet distribution  $Dir(\alpha)$  [271] is used to allocate local samples, ensuring samples of each class are split across

TABLE XI  
EXPERIMENTAL DETAILS. SETTINGS OF DIFFERENT DATASETS IN THE EXPERIMENTS SECTION.

Attributes	CIFAR-10	CIFAR-100	Tiny-ImageNet	Digit-10	Office-31	Office-Caltech-10
Task size	178MB	178MB	435MB	480M	88M	58M
Image number	60K	60K	120K	110K	4.6k	2.5k
Image Size	3×32×32	3×32×32	3×64×64	1×28×28	3×300×300	3×300×300
Task number	$n = 5$	$n = 10$	$n = 10$	$n = 4$	$n = 3$	$n = 4$
Task Scenario	Class-IL	Class-IL	Class-IL	Domain-IL	Domain-IL	Domain-IL
Batch Size	$s = 64$	$s = 64$	$s = 128$	$s = 64$	$s = 32$	$s = 32$
ACC metrics	Top-1	Top-1	Top-10	Top-1	Top-1	Top-1
Learning Rate	$l = 0.01$	$l = 0.01$	$l = 0.001$	$l = 0.001$	$l = 0.01$	$l = 0.01$
Data heterogeneity	$\alpha = 0.1$	$\alpha = 1.0$	$\alpha = 10.0$	$\alpha = 0.1$	$\alpha = 1.0$	$\alpha = 1.0$
Client numbers	$C = 20$	$C = 20$	$C = 20$	$C = 15$	$C = 10$	$C = 8$
Local training epoch	$E = 20$	$E = 20$	$E = 20$	$E = 20$	$E = 20$	$E = 15$
Client selection ratio	$k = 0.4$	$k = 0.5$	$k = 0.6$	$k = 0.4$	$k = 0.4$	$k = 0.5$
Communication Round	$T = 80$	$T = 100$	$T = 100$	$T = 60$	$T = 60$	$T = 40$

TABLE XII  
PERFORMANCE COMPARISON OF VARIOUS METHODS IN TWO INCREMENTAL SCENARIOS.

Method	CIFAR-10		CIFAR-100		Tiny-ImageNet		Digit-10		Office-31		Office-Caltech-10	
	$A(f)$	$\bar{A}$	$A(f)$	$\bar{A}$	$A(f)$	$\bar{A}$	$A(f)$	$\bar{A}$	$A(f)$	$\bar{A}$	$A(f)$	$\bar{A}$
FedAvg	36.68±1.32	59.17±0.08	27.15±0.87	41.36±0.24	30.16±0.19	50.65±0.11	68.12±0.04	80.34±0.02	48.97±0.74	56.29±1.15	55.41±0.52	57.61±0.93
Re-Fed	38.08±0.46	59.02±0.31	32.95±0.31	42.50±0.18	33.43±0.54	51.98±0.32	67.85±0.37	79.85±0.25	50.11±0.29	57.46±0.34	59.16±0.40	60.01±0.33
FedCIL	37.96±1.68	58.30±1.22	30.88±1.04	42.16±0.97	31.35±1.27	50.93±0.84	68.17±0.85	80.02±0.63	49.15±0.92	56.78±0.79	57.80±0.74	59.13±0.52
GLFC	38.43±1.43	60.03±1.16	33.17±0.62	43.28±0.81	32.11±0.40	51.79±0.54	68.12±0.04	80.34±0.02	48.97±0.74	56.29±1.15	55.41±0.52	57.61±0.93
FOT	40.18±1.26	61.41±0.66	36.15±0.40	43.14±0.51	37.23±0.14	54.87±0.22	68.54±0.38	79.70±0.11	49.12±0.83	56.17±0.44	60.30±0.13	60.76±0.98
FedWeIT	37.96±0.52	59.89±0.12	35.84±1.05	44.20±0.45	34.98±0.93	53.04±0.71	69.71±0.20	80.91±0.09	51.49±0.64	57.83±0.89	58.53±0.49	59.72±0.70
CFed	38.19±1.44	60.42±1.32	32.69±0.94	40.30±1.28	32.22±0.86	50.10±0.92	66.89±0.52	77.58±0.76	46.44±0.85	53.29±0.71	54.11±0.59	55.75±0.88
Target	37.68±1.24	58.12±1.63	33.15±1.86	41.05±1.51	29.65±1.41	48.37±1.09	67.19±1.13	76.32±0.85	47.51±1.46	54.82±0.97	55.72±0.65	58.23±0.55
AF-FCL	39.23±1.33	59.78±1.67	34.16±1.15	41.94±0.91	30.95±1.26	50.88±1.10	68.08±0.95	78.65±0.77	48.22±0.86	57.13±0.92	58.81±1.24	60.36±1.05
MFCL	41.26±0.86	62.04±0.77	34.32±1.35	42.79±1.28	31.46±0.84	51.19±0.92	66.33±0.70	75.79±0.76	47.69±0.67	55.24±0.89	56.67±1.05	55.12±0.83
pFedDIL	37.52±0.82	57.29±0.59	34.30±0.61	41.96±0.83	32.87±0.39	50.05±0.42	71.40±0.59	83.80±0.77	53.73±0.35	58.49±0.39	63.13±0.81	63.71±0.72
SR-FDIL	37.98±0.61	58.84±0.80	32.78±0.48	41.55±0.36	31.74±0.78	50.28±0.66	69.86±0.35	81.97±0.24	52.50±0.49	59.21±0.53	60.92±0.70	63.29±0.44

TABLE XIII  
EVALUATION BASED ON THE NUMBER OF COMMUNICATION ROUNDS TO REACH THE BEST ACCURACY. WE USE "Δ" TO REPRESENT THE DIFFERENCE BETWEEN THE PERCENTAGE INCREASE IN ACCURACY AND THE PERCENTAGE INCREASE IN ROUNDS FOR OTHER BASELINES COMPARED TO FEDAVG.

Method	CIFAR-10		CIFAR-100		Tiny-ImageNet		Digit-10		Office-31		Office-Caltech-10	
	Rounds	Δ	Rounds	Δ	Rounds	Δ	Rounds	Δ	Rounds	Δ	Rounds	Δ
FedAvg	304±2.31	/	839±1.67	/	893±2.93	/	208±1.58	/	165±0.94	/	132±1.25	/
Re-Fed	319±1.23	1.03%↓	844±2.15	17.01%↑	894±1.92	9.67%↑	222±1.48	6.70%↓	168±0.98	0.49%↑	145±0.76	2.63%↓
FedCIL	323±2.78	2.51%↓	866±2.25	8.96%↑	903±1.94	2.69%↑	220±1.42	5.38%↓	168±1.07	1.42%↓	147±0.91	6.07%↓
GLFC	312±1.47	1.99%↑	830±2.03	19.23%↑	901±2.12	5.18%↑	220±0.96	6.54%↓	166±1.14	1.99%↓	158±1.31	11.60%↓
FOT	306±2.26	8.06%↑	840±1.35	24.78%↑	903±1.09	17.88%↑	220±0.76	4.84%↓	170±0.98	2.64%↓	158±1.04	8.35%↓
FedWeIT	312±1.95	0.81%↑	870±1.58	20.68%↑	880±2.26	15.26%↑	217±1.31	1.87%↓	165±1.79	4.89%↑	141±1.52	1.05%↓
CFed	317±2.47	0.15%↓	845±2.68	16.24%↑	907±1.54	4.85%↑	225±1.25	9.39%↓	164±0.82	4.84%↓	151±0.58	14.99%↓
Target	310±2.78	0.72%↑	851±3.09	16.69%↑	897±2.31	2.17%↓	220±1.48	6.84%↓	166±1.09	3.68%↓	147±1.70	9.65%↓
AF-FCL	321±2.53	1.20%↑	862±3.21	17.85%↑	911±2.18	0.58%↑	217±1.75	4.21%↓	166±1.31	2.16%↓	145±1.15	3.18%↓
MFCL	320±1.63	6.10%↑	855±2.17	19.02%↑	900±1.93	3.35%↑	223±1.65	9.43%↓	167±0.94	3.88%↓	142±0.87	4.82%↓
pFedDIL	312±1.94	0.33%↓	868±1.88	17.50%↑	908±1.12	6.59%↑	202±0.73	7.56%↑	162±1.25	10.71%↑	140±1.06	6.51%↑
SR-FDIL	314±1.85	0.24%↑	847±2.24	16.23%↑	905±1.36	3.65%↑	205±1.27	3.95%↑	166±1.10	6.12%↑	138±1.34	4.70%↑

clients in different proportions to induce data heterogeneity. A smaller value of  $\alpha$  indicates higher data heterogeneity. To ensure the elimination of variances between baseline methods, unless otherwise specified, we uniformly employ ResNet18 [272] as the backbone model. The implementation of FedCIL and AF-FCL relies on the backbone model itself. To ensure fair comparisons, we set a memory buffer of 300 for each client model to restrict data replay or synthetic samples (e.g., FedAvg can cache 300 samples for each client, and FedCIL also can only generate 300 synthetic samples for local training). Each experimental set is repeated twice, with the average

and standard deviation of accuracy computed over the last 10 rounds. We report the final accuracy  $A(f)$  after completing the last task and the average accuracy  $\bar{A}$  for all tasks. Detailed configurations are presented in Table XI.

#### D. Performance Overview

1) *Test Accuracy*: Table XII compares the test accuracy of various methods on six datasets. While most methods outperform FedAvg, their performance gains remain limited compared to original studies due to constrained memory buffers for fair comparison. In Class-IL tasks, FOT achieves

state-of-the-art performance on Tiny-ImageNet by eliminating data storage requirements through orthogonal projection techniques for parameter isolation. While generator-based methods (e.g., MFCL/AF-FCL) perform well on CIFAR datasets, their effectiveness diminishes significantly on complex datasets like Tiny-ImageNet, indicating limitations in handling large-scale incremental learning. For Domain-IL tasks, regularization methods (e.g., FOT) maintain effectiveness, while data-replay methods struggle under limited memory buffers. Notably, pFedDIL and SR-FDIL excel in domain adaptation but show limited efficacy in Class-IL scenarios. Our analysis reveals no existing method universally addresses both Class-IL and Domain-IL challenges in federated continual learning, highlighting the need for more versatile approaches to handle diverse incremental learning paradigms.

2) *Communication Efficiency*: Table XIII compares communication rounds required by each method to achieve optimal test accuracy, using FedAvg as the baseline for accuracy-efficiency trade-off evaluation. In Class-IL tasks, all methods show positive gains over FedAvg, with accuracy improvements (avg. +10.36%) exceeding added communication costs (avg. +1.97%). For example, MFCL on CIFAR-10 achieves the highest precision despite moderate communication increases. GLFC, FOT, and MFCL exhibit optimal accuracy-efficiency balance. For Domain-IL tasks, Class-IL-oriented methods yield negative  $\Delta$  values as communication costs surpass accuracy gains. In contrast, Domain-IL-specific methods (e.g., pFedDIL, SR-FDIL) maintain low overhead while improving accuracy, achieving positive gains. This highlights task-specific design requirements in federated continual learning.

**Limitations.** In this paper, we simply set a memory buffer for each baseline. However, it should be noted that there are numerous factors that can affect model performance, such as bandwidth, computing overhead, model size, and so on. Achieving absolute uniformity in hardware environments for all methods in large-scale experiments is extremely challenging. We chose to unify the memory buffer based on its significance as a resource factor in NCCL, which has a substantial impact on model performance. In future work, we will gradually expand the benchmark framework to enable the adjustment of different resources.

## VIII. DISCUSSION ABOUT FUTURE RESEARCH DIRECTIONS IN NON-CENTRALIZED CL

Although there is a lot of existing research, there are still challenging new research directions in the deployment of NCCL to be discussed as follows.

- *Computing Saving*: Some existing findings [273] have proved that computational overhead constitutes a pivotal factor in the majority of NCCL methods. However, the acquisition of computational resources, particularly on distributed platforms, frequently incurs substantial costs. Consequently, there is a pressing need to explore more computationally parsimonious algorithms. Furthermore, an alternative perspective revolves around accelerating the convergence of the model, which can significantly reduce the computational cost.

- *Unsupervised Learning*: The existing methods presume full labeling for each task's data, overlooking the reality that real-world situations frequently include both labeled and unlabeled data. Considering the substantial cost of manual annotation, achieving full supervision is unfeasible. Therefore, research into the use of unsupervised or semi-supervised learning within the NCCL framework is crucial, as it can greatly facilitate the practical deployment of NCCL.
- *Resource Allocation*: Given limited training resources, optimizing their allocation to improve model performance is a practical solution. This may involve allocating suitable memory buffers to various data and assigning computational resources to specific processes. One approach to consider is using combinatorial optimization techniques, as suggested by studies such as [274]. In practical applications or method designs, dynamically allocating resources for each technique according to availability can further enhance performance.
- *Online Learning*: Current NCCL research mainly focuses on offline settings, where task data is collected completely before training and remains unchanged. However, in real-world situations, data from ongoing tasks typically arrives in small, incremental batches rather than all at once. Furthermore, the widespread use of resource-limited smart and edge devices requires frequent model updates with incoming data to optimize memory usage and communication efficiency. This presents a difficulty for generative-based methods, as they face challenges in training effective generators with limited data samples.
- *LLM Foundation*: Recently, LLM has attracted considerable attention because of its remarkable abilities, prompting researchers to investigate its integration into NCL frameworks [275]. However, integrating LLM into FIL environments poses a significant challenge. In such settings, distributed devices need to communicate frequently to share knowledge from ongoing tasks, while the server must effectively derive new insights from the LLM. Issues related to communication efficiency and training resources can impede model convergence. Future research should explore innovative approaches for LLM-based NCCL to overcome these obstacles.
- *Generative AI*: Generative AI has emerged as a prominent research area in recent years, as it leverages the generalization capabilities of LLMs to significantly enhance task performance and drive productivity improvements [276]. However, even in centralized learning paradigms, continual learning for Generative AI remains hindered by multiple challenges. Unlike conventional backbone models that primarily face catastrophic forgetting, Generative AI systems more frequently encounter difficulties in acquiring new knowledge [277]. Additionally, the current deployment of Generative AI models through quantization compression on small edge devices complicates continual learning implementation under resource-constrained conditions. While the NCCL has not yet published studies specifically addressing Generative AI, the growing deployment of edge devices equipped with

Generative AI capabilities and their potential to collect real-world data from novel scenarios underscores the urgent need for such research.

- *Convergence Analysis:* Although NCCL research has conducted experiments on various datasets using different methods, empirically demonstrating the effectiveness of the approaches, theoretical research in this area is still lacking. One of the key considerations in NCCL is the convergence of the algorithm. For example, FL aims to find weights that minimize the global model aggregation, which essentially constitutes a distributed optimization problem where convergence is not always guaranteed. Theoretical analysis and evaluations of the convergence bounds of gradient descent-based NCCL for both convex and non-convex loss functions represent important research directions.
- *Generation Task:* Existing NCCL research has primarily focused on classification tasks and achieved considerable success. However, generation tasks are gradually becoming mainstream, such as conversation recommendation [278], chat-bots [279], AI image synthesis [280], and more. Future research needs to expand generation tasks as downstream applications and explore the potential new training challenges they may pose. This will significantly enhance the impact and effectiveness of NCCL, facilitating its practical deployment.
- *Concept Drift:* Concept drift often arises in practical applications, especially in NCCL environments where distributed devices can encounter it. Despite extensive research on traditional concept learning, like macro-financial analysis and epidemic modeling, addressing concept drift in NCCL and contemporary real-world scenarios offers a fascinating and valuable research direction.
- *Asynchronous Learning:* Regarding the current setting of NCCL scenarios, most current NCCL methods focus on synchronous tasks, which assume that incremental tasks arrive at the same time. Asynchronous learning enables each device to collect new training data and initiate training at different timestamps, thereby efficiently utilizing device resources and minimizing unnecessary waiting time. Additionally, asynchronous learning may pose challenges such as global knowledge forgetting and cross-task knowledge integration. However, this more accurately reflects practical NCCL settings and can be a crucial factor in ensuring the scalability of NCCL, given that the NCCL training process can span a longer duration compared to traditional NCCL.

## IX. CONCLUSION

This paper has presented a tutorial on NCCL and a comprehensive survey of the issues related to NCCL implementation. Firstly, we begin with an introduction to the motivation for integrating CL algorithms into the NCL paradigm and how NCCL can serve as a potential technology for better collaborative model training in practical deployment. Then, we describe the fundamentals of NCL paradigms and different CL tasks. Afterward, we provide detailed reviews, analyses, and comparisons of approaches to alleviate catastrophic forgetting and

distribution shifts in NCCL from three levels. Furthermore, we also discuss issues that include heterogeneity, security & privacy, and real-world applications. Finally, we discuss challenges and future research directions.

## ACKNOWLEDGMENTS

This work is supported by the National Key Research and Development Program of China under grant 2024YFC3307900; the National Natural Science Foundation of China under grants 62376103, 62302184, 62436003 and 62206102; Major Science and Technology Project of Hubei Province under grant 2024BAA008; Hubei Science and Technology Talent Service Project under grant 2024DJC078; and Ant Group through CCF-Ant Research Fund. The computation is completed in the HPC Platform of Huazhong University of Science and Technology.

## REFERENCES

- [1] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015. **1**
- [2] A. Koloskova, N. Loizou, S. Boreiri, M. Jaggi, and S. Stich, "A unified theory of decentralized sgd with changing topology and local updates," in *International Conference on Machine Learning*. PMLR, 2020, pp. 5381–5393. **1**
- [3] J.-P. Vergne, "Decentralized vs. distributed organization: Blockchain, machine learning and the future of the digital platform," *Organization Theory*, vol. 1, no. 4, p. 2631787720977052, 2020. **1**
- [4] B. McMahan, E. Moore, D. Ramage, V. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Artificial intelligence and statistics*. PMLR, 2017, pp. 1273–1282. **1, 30**
- [5] J. Kang, Z. Xiong, D. Niyato, Y. Zou, Y. Zhang, and M. Guizani, "Reliable federated learning for mobile networks," *IEEE Wireless Communications*, vol. 27, no. 2, pp. 72–80, 2020. **1**
- [6] M. Xu, H. Du, D. Niyato, J. Kang, Z. Xiong, S. Mao, Z. Han, A. Jamalipour, D. I. Kim, X. Shen, V. C. M. Leung, and H. V. Poor, "Unleashing the power of edge-cloud generative ai in mobile networks: A survey of aigc services," *IEEE Communications Surveys & Tutorials*, vol. 26, no. 2, pp. 1127–1170, 2024. **1**
- [7] Z. Xiong, J. Kang, D. Niyato, P. Wang, and H. V. Poor, "Cloud/edge computing service management in blockchain networks: Multi-leader multi-follower game-based admm for pricing," *IEEE Transactions on Services Computing*, vol. 13, no. 2, pp. 356–367, 2020. **1**
- [8] K. Hsieh, A. Phanishayee, O. Mutlu, and P. Gibbons, "The non-iid data quagmire of decentralized machine learning," in *International Conference on Machine Learning*. PMLR, 2020, pp. 4387–4398. **1**
- [9] B. Liu and Z. Ding, "A consensus-based decentralized training algorithm for deep neural networks with communication compression," *Neurocomputing*, vol. 440, pp. 287–296, 2021. **1**
- [10] P. Sun, X. Liu, Z. Wang, and B. Liu, "Byzantine-robust decentralized federated learning via dual-domain clustering and trust bootstrapping," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 24 756–24 765. **1**
- [11] P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings *et al.*, "Advances and open problems in federated learning," *Foundations and trends® in machine learning*, vol. 14, no. 1–2, pp. 1–210, 2021. **1**
- [12] H. Wang, Y. Li, W. Xu, R. Li, Y. Zhan, and Z. Zeng, "Dafkd: Domain-aware federated knowledge distillation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 20 412–20 421. **1**
- [13] H. Wang, Z. Qu, S. Guo, N. Wang, R. Li, and W. Zhuang, "Losp: Overlap synchronization parallel with local compensation for fast distributed training," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 8, pp. 2541–2557, 2021. **1**
- [14] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: Vision and challenges," *IEEE internet of things journal*, vol. 3, no. 5, pp. 637–646, 2016. **1**

- [15] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A survey on mobile edge computing: The communication perspective," *IEEE communications surveys & tutorials*, vol. 19, no. 4, pp. 2322–2358, 2017. **1**
- [16] Z. Zhong, W. Bao, J. Wang, X. Zhu, and X. Zhang, "Flee: A hierarchical federated learning framework for distributed deep neural network over cloud, edge, and end device," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 13, no. 5, pp. 1–24, 2022. **1**
- [17] Y. Li, Y. Shan, Y. LIU, H. Wang, wangshi.w, Y. Wang, and R. Li, "Personalized federated recommendation for cold-start users via adaptive knowledge fusion," in *THE WEB CONFERENCE 2025*, 2025. [Online]. Available: <https://openreview.net/forum?id=bhWngwu74> **1, 13**
- [18] Y. Li, Q. Qin, G. Zhu, W. Xu, H. Wang, Y. Li, R. Zhang, and R. Li, "A systematic survey on federated sequential recommendation," *Preprints*, March 2025. [Online]. Available: <https://doi.org/10.20944/preprints202503.1015.v1> **1**
- [19] X. Zheng, Z. Wang, C. Chen, J. Qian, and Y. Yang, "Decentralized graph neural network for privacy-preserving recommendation," in *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, 2023, pp. 3494–3504. **1**
- [20] J. Kang, R. Yu, X. Huang, S. Maharjan, Y. Zhang, and E. Hossain, "Enabling localized peer-to-peer electricity trading among plug-in hybrid electric vehicles using consortium blockchains," *IEEE transactions on industrial informatics*, vol. 13, no. 6, pp. 3154–3164, 2017. **1**
- [21] J. Kang, Z. Xiong, D. Niyato, D. Ye, D. I. Kim, and J. Zhao, "Toward secure blockchain-enabled internet of vehicles: Optimizing consensus management using reputation and contract theory," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 3, pp. 2906–2920, 2019. **1**
- [22] M. Xu, D. Niyato, J. Chen, H. Zhang, J. Kang, Z. Xiong, S. Mao, and Z. Han, "Generative ai-empowered simulation for autonomous driving in vehicular mixed reality metaverses," *IEEE Journal of Selected Topics in Signal Processing*, vol. 17, no. 5, pp. 1064–1079, 2023. **1**
- [23] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska et al., "Overcoming catastrophic forgetting in neural networks," *Proceedings of the national academy of sciences*, vol. 114, no. 13, pp. 3521–3526, 2017. **2**
- [24] A. Robins, "Catastrophic forgetting, rehearsal and pseudorehearsal," *Connection Science*, vol. 7, no. 2, pp. 123–146, 1995. **2**
- [25] R. M. French, "Catastrophic forgetting in connectionist networks," *Trends in cognitive sciences*, vol. 3, no. 4, pp. 128–135, 1999. **2**
- [26] M. Ye, X. Fang, B. Du, P. C. Yuen, and D. Tao, "Heterogeneous federated learning: State-of-the-art and research challenges," *ACM Computing Surveys*, vol. 56, no. 3, pp. 1–44, 2023. **2**
- [27] H. Ning, Y. Li, F. Shi, and L. T. Yang, "Heterogeneous edge computing open platforms and tools for internet of things," *Future Generation Computer Systems*, vol. 106, pp. 67–76, 2020. **2**
- [28] C. Zhang, Y. Xie, H. Bai, B. Yu, W. Li, and Y. Gao, "A survey on federated learning," *Knowledge-Based Systems*, vol. 216, p. 106775, 2021. **2, 4**
- [29] W. Y. B. Lim, N. C. Luong, D. T. Hoang, Y. Jiao, Y.-C. Liang, Q. Yang, D. Niyato, and C. Miao, "Federated learning in mobile edge networks: A comprehensive survey," *IEEE communications surveys & tutorials*, vol. 22, no. 3, pp. 2031–2063, 2020. **2, 4**
- [30] L. U. Khan, W. Saad, Z. Han, E. Hossain, and C. S. Hong, "Federated learning for internet of things: Recent advances, taxonomy, and open challenges," *IEEE Communications Surveys & Tutorials*, vol. 23, no. 3, pp. 1759–1799, 2021. **2**
- [31] A. Imteaj, U. Thakker, S. Wang, J. Li, and M. H. Amini, "A survey on federated learning for resource-constrained iot devices," *IEEE Internet of Things Journal*, vol. 9, no. 1, pp. 1–24, 2021. **2**
- [32] E. T. M. Beltrán, M. Q. Pérez, P. M. S. Sánchez, S. L. Bernal, G. Bovet, M. G. Pérez, G. M. Pérez, and A. H. Celdrán, "Decentralized federated learning: Fundamentals, state of the art, frameworks, trends, and challenges," *IEEE Communications Surveys & Tutorials*, 2023. **2**
- [33] M. De Lange, R. Aljundi, M. Masana, S. Parisot, X. Jia, A. Leonardis, G. Slabaugh, and T. Tuytelaars, "A continual learning survey: Defying forgetting in classification tasks," *IEEE transactions on pattern analysis and machine intelligence*, vol. 44, no. 7, pp. 3366–3385, 2021. **2, 4**
- [34] G. M. Van de Ven, T. Tuytelaars, and A. S. Tolias, "Three types of incremental learning," *Nature Machine Intelligence*, vol. 4, no. 12, pp. 1185–1197, 2022. **2, 4**
- [35] M. Masana, X. Liu, B. Twardowski, M. Menta, A. D. Bagdanov, and J. Van De Weijer, "Class-incremental learning: survey and performance evaluation on image classification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 5, pp. 5513–5533, 2022. **2, 4**
- [36] Y. Huang, C. Bert, A. Goma, R. Fietkau, A. Maier, and F. Putz, "A survey of incremental transfer learning: Combining peer-to-peer federated learning and domain incremental learning for multicenter collaboration," *arXiv preprint arXiv:2309.17192*, 2023. **2, 4**
- [37] M. F. Criado, F. E. Casado, R. Iglesias, C. V. Regueiro, and S. Barro, "Non-iid data and continual learning processes in federated learning: A long road ahead," *Information Fusion*, vol. 88, pp. 263–280, 2022. **2, 4**
- [38] X. Yang, H. Yu, X. Gao, H. Wang, J. Zhang, and T. Li, "Federated continual learning via knowledge fusion: A survey," *IEEE Transactions on Knowledge and Data Engineering*, 2024. **2, 3, 4**
- [39] Q. Wang, B. Liu, and Y. Li, "Traceable federated continual learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 12 872–12 881. **3**
- [40] J. Yoon, W. Jeong, G. Lee, E. Yang, and S. J. Hwang, "Federated continual learning with weighted inter-client transfer," in *International Conference on Machine Learning*. PMLR, 2021, pp. 12 073–12 086. **3, 30**
- [41] Y. Ma, Z. Xie, J. Wang, K. Chen, L. Shou, and L. De Raedt, "Continual federated learning based on knowledge distillation," in *IJCAI*, 2022, pp. 2182–2188. **3, 8, 17, 27, 28, 29**
- [42] D. Qi, H. Zhao, and S. Li, "Better generative replay for continual federated learning," *arXiv preprint arXiv:2302.13001*, 2023. **3, 8, 10, 30**
- [43] J. Zhang, C. Chen, W. Zhuang, and L. Lyu, "Target: Federated class-continual learning via exemplar-free distillation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 4782–4793. **3, 10, 30**
- [44] A. Wuerkaixi, S. Cui, J. Zhang, K. Yan, B. Han, G. Niu, L. Fang, C. Zhang, and M. Sugiyama, "Accurate forgetting for heterogeneous federated continual learning," in *The Twelfth International Conference on Learning Representations*, 2024. **3, 10, 28, 29, 30**
- [45] Y. Li, W. Xu, Y. Qi, H. Wang, R. Li, and S. Guo, "Sr-fdil: Synergistic replay for federated domain-incremental learning," *IEEE Transactions on Parallel and Distributed Systems*, vol. 35, no. 11, pp. 1879–1890, 2024. **3, 8, 10, 14, 15, 27, 28, 30**
- [46] Y. Li, Q. Li, H. Wang, R. Li, W. Zhong, and G. Zhang, "Towards efficient replay in federated incremental learning," *arXiv preprint arXiv:2403.05890*, 2024. **3, 9, 27, 28, 30**
- [47] Y. Li, H. Wang, Y. Qi, W. Liu, and R. Li, "Re-fed+: A better replay strategy for federated incremental learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–12, 2025. **3, 8**
- [48] J. Dong, D. Zhang, Y. Cong, W. Cong, H. Ding, and D. Dai, "Federated incremental semantic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2023, pp. 3934–3943. **3**
- [49] J. Dong, H. Li, Y. Cong, G. Sun, Y. Zhang, and L. Van Gool, "No one left behind: Real-world federated class-incremental learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023. **3, 28**
- [50] F. E. Casado, D. Lema, R. Iglesias, C. V. Regueiro, and S. Barro, "Federated and continual learning for classification tasks in a society of devices," *arXiv preprint arXiv:2006.07129*, 2020. **3**
- [51] C. Lanza, E. Angelats, M. Miozzo, and P. Dini, "Urban traffic forecasting using federated and continual learning," in *2023 6th Conference on Cloud and Internet of Things (CIoT)*. IEEE, 2023, pp. 1–8. **3**
- [52] J. Verbraeken, M. Wolting, J. Katzy, J. Kloppenburg, T. Verbelen, and J. S. Rellermeyer, "A survey on distributed machine learning," *Acm computing surveys (csur)*, vol. 53, no. 2, pp. 1–33, 2020. **3**
- [53] H. T. Dinh, C. Lee, D. Niyato, and P. Wang, "A survey of mobile cloud computing: architecture, applications, and approaches," *Wireless communications and mobile computing*, vol. 13, no. 18, pp. 1587–1611, 2013. **3**
- [54] H. Wang, Y. Jia, M. Zhang, Q. Hu, H. Ren, P. Sun, Y. Wen, and T. Zhang, "Fedds: Distribution-aware sub-model extraction for federated learning over resource-constrained devices," in *Proceedings of the ACM on Web Conference 2024*, 2024, pp. 2902–2913. **3, 5**
- [55] G. Sun, Y. Cong, J. Dong, Q. Wang, L. Lyu, and J. Liu, "Data poisoning attacks on federated machine learning," *IEEE Internet of Things Journal*, vol. 9, no. 13, pp. 11 365–11 375, 2022. **4**
- [56] W. Liu, L. Chen, and W. Zhang, "Decentralized federated learning: Balancing communication and computing costs," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 8, pp. 131–143, 2022. **5**

- [57] G. B. Giannakis, Q. Ling, G. Mateos, I. D. Schizas, and H. Zhu, "Decentralized learning for wireless communications and networking," in *Splitting Methods in Communication, Imaging, Science, and Engineering*. Springer, 2017, pp. 461–497. 5
- [58] K. Hsieh, A. Phanishayee, O. Mutlu, and P. Gibbons, "The non-iid data quagmire of decentralized machine learning," in *International Conference on Machine Learning*. PMLR, 2020, pp. 4387–4398. 5
- [59] S. Li, T. Zhou, X. Tian, and D. Tao, "Learning to collaborate in decentralized learning of personalized models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 9766–9775. 5
- [60] A. Elgabli, J. Park, A. S. Bedi, M. Bennis, and V. Aggarwal, "Gadmm: Fast and communication efficient framework for distributed machine learning," *Journal of Machine Learning Research*, vol. 21, no. 76, pp. 1–39, 2020. 5
- [61] P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings *et al.*, "Advances and open problems in federated learning," *Foundations and trends® in machine learning*, vol. 14, no. 1–2, pp. 1–210, 2021. 5
- [62] H. Wang, P. Zheng, X. Han, W. Xu, R. Li, and T. Zhang, "Fednlr: Federated learning with neuron-wise learning rates," in *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2024, pp. 3069–3080. 5
- [63] H. Wang, H. Xu, Y. Li, Y. Xu, R. Li, and T. Zhang, "Fedcda: Federated learning with cross-rounds divergence-aware aggregation," in *The Twelfth International Conference on Learning Representations*, 2023. 5
- [64] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Artificial intelligence and statistics*. PMLR, 2017, pp. 1273–1282. 5
- [65] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," *Proceedings of Machine learning and systems*, vol. 2, pp. 429–450, 2020. 5
- [66] Q. Li, B. He, and D. Song, "Model-contrastive federated learning," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 10713–10722. 5
- [67] S. P. Karimireddy, S. Kale, M. Mohri, S. Reddi, S. Stich, and A. T. Suresh, "Scaffold: Stochastic controlled averaging for federated learning," in *International conference on machine learning*. PMLR, 2020, pp. 5132–5143. 5
- [68] J. Liu, X. Wei, X. Liu, H. Gao, and Y. Wang, "Group-based hierarchical federated learning: Convergence, group formation, and sampling," in *Proceedings of the 52nd International Conference on Parallel Processing*, 2023, pp. 264–273. 6
- [69] W. Y. B. Lim, J. S. Ng, Z. Xiong, D. Niyato, C. Miao, and D. I. Kim, "Dynamic edge association and resource allocation in self-organizing hierarchical federated learning networks," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 12, pp. 3640–3653, 2021. 6
- [70] S. Luo, X. Chen, Q. Wu, Z. Zhou, and S. Yu, "Hfel: Joint edge association and resource allocation for cost-efficient hierarchical federated edge learning," *IEEE Transactions on Wireless Communications*, vol. 19, no. 10, pp. 6535–6548, 2020. 6
- [71] Q. Chen, Z. You, D. Wen, and Z. Zhang, "Enhanced hybrid hierarchical federated edge learning over heterogeneous networks," *IEEE Transactions on Vehicular Technology*, vol. 72, no. 11, pp. 14601–14614, 2023. 6
- [72] Z. Li, J. Lu, S. Luo, D. Zhu, Y. Shao, Y. Li, Z. Zhang, Y. Wang, and C. Wu, "Towards effective clustered federated learning: A peer-to-peer framework with adaptive neighbor matching," *IEEE Transactions on Big Data*, 2022. 6
- [73] K. Hu, J. Wu, L. Weng, Y. Zhang, F. Zheng, Z. Pang, and M. Xia, "A novel federated learning approach based on the confidence of federated kalman filters," *International Journal of Machine Learning and Cybernetics*, vol. 12, pp. 3607–3627, 2021. 6
- [74] W. Li and S. Wang, "Federated meta-learning for spatial-temporal prediction," *Neural Computing and Applications*, vol. 34, no. 13, pp. 10355–10374, 2022. 6
- [75] X. Wang, A. Lalitha, T. Javidi, and F. Koushanfar, "Peer-to-peer variational federated learning over arbitrary graphs," *IEEE Journal on Selected Areas in Information Theory*, vol. 3, no. 2, pp. 172–182, 2022. 6
- [76] Y. Xu, C. Peng, W. Tan, Y. Tian, M. Ma, and K. Niu, "Non-interactive verifiable privacy-preserving federated learning," *Future Generation Computer Systems*, vol. 128, pp. 365–380, 2022. 6
- [77] J. Huang, C. Xu, Z. Ji, S. Xiao, T. Liu, N. Ma, and Q. Zhou, "[retracted] aflpc: An asynchronous federated learning privacy-preserving computing model applied to 5g-v2x," *Security and Communication Networks*, vol. 2022, no. 1, p. 9334943, 2022. 6
- [78] J. Park and H. Lim, "Privacy-preserving federated learning using homomorphic encryption," *Applied Sciences*, vol. 12, no. 2, p. 734, 2022. 6
- [79] A. Gholami, N. Torkzaban, and J. S. Baras, "Trusted decentralized federated learning," in *2022 IEEE 19th Annual Consumer Communications & Networking Conference (CCNC)*. IEEE, 2022, pp. 1–6. 6
- [80] C. Dwork, A. Roth *et al.*, "The algorithmic foundations of differential privacy," *Foundations and Trends® in Theoretical Computer Science*, vol. 9, no. 3–4, pp. 211–407, 2014. 6
- [81] A. Imteaj, K. Mamun Ahmed, U. Thakker, S. Wang, J. Li, and M. H. Amini, "Federated learning for resource-constrained iot devices: Panoramas and state of the art," *Federated and Transfer Learning*, pp. 7–27, 2022. 6
- [82] Z. Li, Y. Zhou, D. Wu, T. Tang, and R. Wang, "Fairness-aware federated learning with unreliable links in resource-constrained internet of things," *IEEE Internet of Things Journal*, vol. 9, no. 18, pp. 17359–17371, 2022. 6
- [83] Z. Chen and B. Liu, *Lifelong machine learning*. Morgan & Claypool Publishers, 2018. 6
- [84] G. I. Parisi, R. Kemker, J. L. Part, C. Kanan, and S. Wermter, "Continual lifelong learning with neural networks: A review," *Neural networks*, vol. 113, pp. 54–71, 2019. 6
- [85] R. Aljundi, P. Chakravarty, and T. Tuytelaars, "Expert gate: Lifelong learning with a network of experts," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 3366–3375. 6
- [86] A. Gepperth and C. Karaoguz, "A bio-inspired incremental learning architecture for applied perceptual problems," *Cognitive Computation*, vol. 8, no. 5, pp. 924–934, 2016. 6
- [87] A. Rosenfeld and J. K. Tsotsos, "Incremental learning through deep adaptation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 42, no. 3, pp. 651–663, 2018. 6, 7
- [88] S.-A. Rebuffi, A. Kolesnikov, G. Sperl, and C. H. Lampert, "icarl: Incremental classifier and representation learning," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2017, pp. 2001–2010. 6
- [89] D. Lopez-Paz and M. Ranzato, "Gradient episodic memory for continual learning," *Advances in neural information processing systems*, vol. 30, 2017. 6
- [90] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska *et al.*, "Overcoming catastrophic forgetting in neural networks," *Proceedings of the national academy of sciences*, vol. 114, no. 13, pp. 3521–3526, 2017. 7, 13
- [91] F. Zenke, B. Poole, and S. Ganguli, "Continual learning through synaptic intelligence," in *International conference on machine learning*. PMLR, 2017, pp. 3987–3995. 7
- [92] Z. Li and D. Hoiem, "Learning without forgetting," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 12, pp. 2935–2947, 2017. 7
- [93] J. Serra, D. Suris, M. Miron, and A. Karatzoglou, "Overcoming catastrophic forgetting with hard attention to the task," in *International conference on machine learning*. PMLR, 2018, pp. 4548–4557. 7
- [94] D. Li, Z. Chen, E. Cho, J. Hao, X. Liu, F. Xing, C. Guo, and Y. Liu, "Overcoming catastrophic forgetting during domain adaptation of seq2seq language generation," in *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2022, pp. 5441–5454. 7
- [95] V. Varshney, M. Patidar, R. Kumar, G. Shroff, and L. Vig, "Prompt augmented generative replay via supervised contrastive training for lifelong intent detection," Jan. 11 2024, uS Patent App. 18/215,972. 7
- [96] T. Scialom, T. Chakrabarty, and S. Muresan, "Fine-tuned language models are continual learners," *arXiv preprint arXiv:2205.12393*, 2022. 7
- [97] A. Prabhu, H. A. Al Kader Hammoud, P. K. Dokania, P. H. Torr, S.-N. Lim, B. Ghanem, and A. Bibi, "Computationally budgeted continual learning: What does matter?" in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 3698–3707. 7
- [98] G. M. Van de Ven and A. S. Tolias, "Generative replay with feedback connections as a general strategy for continual learning," *arXiv preprint arXiv:1809.10635*, 2018. 7

- [99] B. Cox, J. Galjaard, A. Shankar, J. Decouchant, and L. Y. Chen, "Parameterizing federated continual learning for reproducible research," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 2023, pp. 478–486. [8](#)
- [100] F. Wu, S. Feng, Y. Chen, and L. Zhao, "Personalized federated class-incremental learning through critical parameter transfer," in *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2025, pp. 1–5. [8](#)
- [101] Y. Li, W. Xu, H. Wang, R. Li, Y. Qi, and J. Guo, "Personalized federated domain-incremental learning based on adaptive knowledge matching," 2024. [Online]. Available: <https://arxiv.org/abs/2407.05005> [8](#), [30](#)
- [102] C. Chen, I. Kevin, K. Wang, P. Li, and K. Sakurai, "Flexibility and privacy: A multi-head federated continual learning framework for dynamic edge environments," in *2023 Eleventh International Symposium on Computing and Networking (CANDAR)*. IEEE, 2023, pp. 1–10. [8](#)
- [103] Y. He, C. Shen, X. Wang, and B. Jin, "Fppl: An efficient and non-iid robust federated continual learning framework," *arXiv preprint arXiv:2411.01904*, 2024. [8](#), [12](#), [14](#)
- [104] J. Yoon and H. Lee, "Pick-a-back: Selective device-to-device knowledge transfer in federated continual learning," in *European Conference on Computer Vision*. Springer, 2025, pp. 165–182. [8](#), [15](#), [16](#)
- [105] C. Gong, Z. Zheng, F. Wu, X. Jia, and G. Chen, "Delta: A cloud-assisted data enrichment framework for on-device continual learning," *arXiv preprint arXiv:2410.18378*, 2024. [9](#)
- [106] K. Alhamoud, Y. Ghunaim, M. Alfarrar, T. Hartvigsen, P. Torr, B. Ghanem, A. Bibi, and M. Ghassemi, "Fedmedicl: Towards holistic evaluation of distribution shifts in federated medical imaging," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2024, pp. 383–393. [9](#)
- [107] S. Dahdal, A. Gilli, F. Poltronieri, M. Tortonesi, C. Stefanelli, and N. Suri, "Roamml platform: Enabling distributed continual learning for disaster relief operations," in *2024 IEEE Symposium on Computers and Communications (ISCC)*. IEEE, 2024, pp. 1–6. [9](#)
- [108] M. Revathi *et al.*, "Exploring the efficacy of federated-continual learning nodes with attention-based classifier for robust web phishing detection: An empirical investigation," *arXiv preprint arXiv:2405.03537*, 2024. [9](#)
- [109] W. Zhuang, J. Xu, C. Chen, J. Li, and L. Lyu, "Coala: A practical and vision-centric federated learning platform," *arXiv preprint arXiv:2407.16560*, 2024. [9](#)
- [110] G. Serra and F. Buettner, "Federated continual learning goes online: Leveraging uncertainty for modality-agnostic class-incremental learning," *arXiv preprint arXiv:2405.18925*, 2024. [9](#)
- [111] L. Le, M. Hussing, and E. Eaton, "Distributed continual learning," *arXiv preprint arXiv:2405.17466*, 2024. [9](#)
- [112] Y. Mei, L. Yuan, D.-J. Han, K. S. Chan, C. G. Brinton, and T. Lan, "Using diffusion models as generative replay in continual federated learning—what will happen?" *arXiv preprint arXiv:2411.06618*, 2024. [9](#), [17](#)
- [113] C. Zhang, T. Huang, W. Mao, H. Bai, and B. Yu, "Federated continual learning based on central memory rehearsal," *Hospital*, vol. 1, p. T2. [9](#), [17](#)
- [114] B. Li, P. Song, C. Zhao, and M. Xie, "Facing spatiotemporal heterogeneity: A unified federated continual learning framework with self-challenge rehearsal for industrial monitoring tasks," *Knowledge-Based Systems*, vol. 289, p. 111491, 2024. [9](#)
- [115] X. Liu, S. Wang, and Y. Chen, "Adaptive scheduling for internet of vehicles using deconfounded graph transfer learning," *Computer Networks*, p. 110899, 2024. [9](#)
- [116] M. K. Yoo and Y. R. Park, "Federated class incremental learning: A pseudo feature based approach without exemplars," in *Proceedings of the Asian Conference on Computer Vision*, 2024, pp. 488–498. [9](#)
- [117] N. Churamani, S. Checker, H.-T. L. Chiang, and H. Gunes, "Feature aggregation with latent generative replay for federated continual learning of socially appropriate robot behaviours," *arXiv preprint arXiv:2405.15773*, 2024. [9](#)
- [118] Z. Wu, T. He, S. Sun, Y. Wang, M. Liu, B. Gao, and X. Jiang, "Federated class-incremental learning with new-class augmented self-distillation," *arXiv preprint arXiv:2401.00622*, 2024. [9](#), [11](#)
- [119] A. Usmanova, F. Portet, P. Lalanda, and G. Vega, "Federated continual learning through distillation in pervasive computing," in *2022 IEEE International Conference on Smart Computing (SMARTCOMP)*, 2022, pp. 86–91. [9](#), [11](#), [23](#)
- [120] N. Babendererde, H. Zhu, M. Fuchs, J. Stieber, and A. Mukhopadhyay, "Federated-continual dynamic segmentation of histopathology guided by barlow continuity," *arXiv preprint arXiv:2501.04588*, 2025. [9](#)
- [121] J. Xie, C. Zhu, and S. Li, "Fedmes: Personalized federated continual learning leveraging local memory," *arXiv preprint arXiv:2404.12710*, 2024. [8](#)
- [122] S. Sebaa, M. Mabrouki, Y. Ghamri-Doudane, K. Benatchba *et al.*, "A life-long learning intrusion detection system for 6g-enabled iot," *arXiv preprint arXiv:2407.15700*, 2024. [9](#)
- [123] D. Rolnick, A. Ahuja, J. Schwarz, T. Lillicrap, and G. Wayne, "Experience replay for continual learning," *Advances in neural information processing systems*, vol. 32, 2019. [9](#)
- [124] A. Odena, C. Olah, and J. Shlens, "Conditional image synthesis with auxiliary classifier gans," in *International conference on machine learning*. PMLR, 2017, pp. 2642–2651. [10](#)
- [125] M. Khademi Nori, I.-M. Kim, and G. Wang, "Federated class-incremental learning: A hybrid approach using latent exemplars and data-free techniques to address local and global forgetting," *arXiv e-prints*, pp. arXiv–2501, 2025. [10](#)
- [126] R. Sun, Y. Zhang, V. Ojha, T. Shah, H. Duan, B. Wei, and R. Ranjan, "Exemplar-condensed federated class-incremental learning," *arXiv preprint arXiv:2412.18926*, 2024. [10](#)
- [127] S. Babakniya, Z. Fabian, C. He, M. Soltanolkotabi, and S. Avestimehr, "A data-free approach to mitigate catastrophic forgetting in federated class incremental learning for vision tasks," in *Proceedings of the 37th International Conference on Neural Information Processing Systems*, ser. NIPS '23. Red Hook, NY, USA: Curran Associates Inc., 2024. [10](#), [30](#)
- [128] R. Salami, P. Buzzega, M. Mosconi, M. Verasani, and S. Calderara, "Federated class-incremental learning with hierarchical generative prototypes," *arXiv preprint arXiv:2406.02447*, 2024. [10](#)
- [129] H. Miao, Y. Zhao, C. Guo, B. Yang, K. Zheng, and C. S. Jensen, "Spatio-temporal prediction on streaming data: A unified federated continuous learning framework," *IEEE Transactions on Knowledge and Data Engineering*, 2025. [11](#)
- [130] K. Gai, Z. Wang, J. Yu, and L. Zhu, "Mufti: Multi-domain distillation-based heterogeneous federated continuous learning," *IEEE Transactions on Information Forensics and Security*, 2025. [11](#)
- [131] L. Yu, L. Ge, G. Wang, J. Yin, Q. Chen, J. Zhou, and L. He, "Pi-fed: Continual federated learning with parameter-level importance aggregation," *IEEE Internet of Things Journal*, 2024. [12](#)
- [132] H. Yu, X. Yang, X. Gao, Y. Feng, H. Wang, Y. Kang, and T. Li, "Overcoming spatial-temporal catastrophic forgetting for federated class-incremental learning," in *Proceedings of the 32nd ACM International Conference on Multimedia*, 2024, pp. 5280–5288. [12](#)
- [133] A. Z. Tan, S. Feng, and H. Yu, "Fl-clip: Bridging plasticity and stability in pre-trained federated class-incremental learning models," in *2024 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 2024, pp. 1–6. [12](#)
- [134] S. Wang, K. Gai, J. Yu, L. Zhu, and Q. Wu, "Vertical federated continual learning via evolving prototype knowledge," *arXiv preprint arXiv:2502.09152*, 2025. [12](#)
- [135] H. Yu, X. Yang, L. Zhang, H. Gu, T. Li, L. Fan, and Q. Yang, "Addressing spatial-temporal data heterogeneity in federated continual learning via tail anchor," *arXiv preprint arXiv:2412.18355*, 2024. [12](#)
- [136] H. Xu, "Federated continual learning based on weight self-optimization algorithm in non-iid," in *Third International Conference on Electronic Information Engineering and Data Processing (EIEDP 2024)*, vol. 13184. SPIE, 2024, pp. 872–878. [12](#)
- [137] Z. Zhao, F. Lyu, L. Li, F. Hu, M. Gu, and L. Sun, "Towards long-term remembering in federated continual learning," *Cognitive Computation*, pp. 1–9, 2024. [12](#)
- [138] H. Wang, J. Sun, T. Wo, and X. Liu, "Fedfr: Federated forgetting-resistant representation learning," in *2024 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 2024, pp. 1–6. [12](#)
- [139] H. Yuan, X. He, R. Hu, Z. Wang, J. Zhou, Y. Feng, and L. Gong, "Gcgn: A group connected based method for continual vertical federated recommender systems in data ecosystems," in *2024 IEEE International Conference on Web Services (ICWS)*. IEEE, 2024, pp. 35–44. [12](#)
- [140] Z. Yang, S. Zhang, C. Li, H. Wang, and M. Zhang, "Resource-efficient heterogeneous federated continual learning on edge," in *2024 Design, Automation & Test in Europe Conference & Exhibition (DATE)*. IEEE, 2024, pp. 1–6. [12](#)
- [141] Y. He and X. Wang, "Masked autoencoders are parameter-efficient federated continual learners," *arXiv preprint arXiv:2411.01916*, 2024. [12](#)
- [142] R. Salami, P. Buzzega, M. Mosconi, J. Bonato, L. Sabetta, and S. Calderara, "Closed-form merging of parameter-efficient modules for federated continual learning," *arXiv preprint arXiv:2410.17961*, 2024. [12](#)

- [143] R. Salami, P. Buzzega, M. Mosconi, M. Verasani, and S. Calderara, "Reducing bias in federated class-incremental learning with hierarchical generative prototypes," *arXiv preprint arXiv:2406.02447*, 2024. **12**
- [144] D. Peng, Y. Wang, H. Fu, Q. Wei, Y. Liu, and R. S. M. Goh, "Learning task-specific initialization for effective federated continual fine-tuning of foundation model adapters," in *2024 IEEE Conference on Artificial Intelligence (CAI)*. IEEE, 2024, pp. 811–816. **12**
- [145] H. Ritter, A. Botev, and D. Barber, "Online structured laplace approximations for overcoming catastrophic forgetting," *Advances in Neural Information Processing Systems*, vol. 31, 2018. **11**
- [146] J. Schwarz, W. Czarniecki, J. Luketina, A. Grabska-Barwinska, Y. W. Teh, R. Pascanu, and R. Hadsell, "Progress & compress: A scalable framework for continual learning," in *International conference on machine learning*. PMLR, 2018, pp. 4528–4537. **11**
- [147] J. Feng, X. Yang, L. Liang, W. Han, B. Fang, and Q. Liao, "Cgofed: Constrained gradient optimization strategy for federated class incremental learning," *IEEE Transactions on Knowledge and Data Engineering*, 2025. **11**
- [148] Y. F. Bakman, D. N. Yaldiz, Y. H. Ezzeldin, and S. Avestimehr, "Federated orthogonal training: Mitigating global catastrophic forgetting in continual federated learning," *arXiv preprint arXiv:2309.01289*, 2023. **12, 30**
- [149] G. Saha, I. Garg, and K. Roy, "Gradient projection memory for continual learning," *arXiv preprint arXiv:2103.09762*, 2021. **12**
- [150] N. Shoham, T. Avidor, A. Keren, N. Israel, D. Benditkis, L. Mor-Yosef, and I. Zeitak, "Overcoming forgetting in federated learning on non-iid data," *arXiv preprint arXiv:1910.07796*, 2019. **13**
- [151] R. Hadsell, D. Rao, A. A. Rusu, and R. Pascanu, "Embracing change: Continual learning in deep neural networks," *Trends in cognitive sciences*, vol. 24, no. 12, pp. 1028–1040, 2020. **13**
- [152] A. A. Rusu, N. C. Rabinowitz, G. Desjardins, H. Soyer, J. Kirkpatrick, K. Kavukcuoglu, R. Pascanu, and R. Hadsell, "Progressive neural networks," *arXiv preprint arXiv:1606.04671*, 2016. **13**
- [153] J. Xu and Z. Zhu, "Reinforced continual learning," *Advances in neural information processing systems*, vol. 31, 2018. **13**
- [154] J. Le, X. Lei, N. Mu, H. Zhang, K. Zeng, and X. Liao, "Federated continuous learning with broad network architecture," *IEEE Transactions on Cybernetics*, vol. 51, no. 8, pp. 3874–3888, 2021. **13, 28, 29**
- [155] X. Dong, Z. Yu, W. Cao, Y. Shi, and Q. Ma, "A survey on ensemble learning. frontiers of computer science, 14: 241-258," 2020. **13**
- [156] L. Yuan, Y. Chen, T. Wang, W. Yu, Y. Shi, Z.-H. Jiang, F. E. Tay, J. Feng, and S. Yan, "Tokens-to-token vit: Training vision transformers from scratch on imagenet," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 558–567. **13**
- [157] X. Zhai, A. Kolesnikov, N. Houlsby, and L. Beyer, "Scaling vision transformers," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 12 104–12 113. **13**
- [158] B. Lester, R. Al-Rfou, and N. Constant, "The power of scale for parameter-efficient prompt tuning," *arXiv preprint arXiv:2104.08691*, 2021. **14**
- [159] S. Paul and P.-Y. Chen, "Vision transformers are robust learners," in *Proceedings of the AAAI conference on Artificial Intelligence*, vol. 36, no. 2, 2022, pp. 2071–2081. **14**
- [160] H. Yu, X. Yang, X. Gao, Y. Kang, H. Wang, J. Zhang, and T. Li, "Personalized federated continual learning via multi-granularity prompt," *arXiv preprint arXiv:2407.00113*, 2024. **14, 17**
- [161] H. Piao, Y. Wu, D. Wu, and Y. Wei, "Federated continual learning via prompt-based dual knowledge transfer," in *Forty-first International Conference on Machine Learning*. **14**
- [162] H. Guo, F. Zhu, W. Liu, X.-Y. Zhang, and C.-L. Liu, "Federated class-incremental learning with prototype guided transformer," *arXiv preprint arXiv:2401.02094*, 2024. **14**
- [163] X. Jiang and C. Borcea. Concept matching: Clustering-based federated continual learning. [Online]. Available: <http://arxiv.org/abs/2311.06921> **15**
- [164] L. Fu, H. Zhang, G. Gao, M. Zhang, and X. Liu, "Client selection in federated learning: Principles, challenges, and opportunities," *IEEE Internet of Things Journal*, 2023. **14**
- [165] J. Yoon and H. Lee, "Pick-a-back: Selective device-to-device knowledge transfer in federated continual learning," in *European Conference on Computer Vision*. Springer, 2024, pp. 165–182. **15**
- [166] G. Long, M. Xie, T. Shen, T. Zhou, X. Wang, and J. Jiang, "Multi-center federated learning: clients clustering for better personalization," *World Wide Web*, vol. 26, no. 1, pp. 481–500, 2023. **15**
- [167] L. Palazzo, M. Pennisi, F. P. Salanitri, G. Bellitto, S. Palazzo, and C. Spampinato, "Fedrewind: Rewinding continual model exchange for decentralized federated learning," *arXiv preprint arXiv:2411.09842*, 2024. **16**
- [168] F. E. Casado, D. Lema, R. Iglesias, C. V. Regueiro, and S. Barro, "Federated and continual learning for classification tasks in a society of devices," *arXiv preprint arXiv:2006.07129*, 2020. **16, 18**
- [169] Z. Song, X. Yang, Y. Zhang, X. Fu, Z. Xu, and I. King, "A systematic survey on federated semi-supervised learning," in *IJCAI*, 2024. **16, 18**
- [170] W. Huang, M. Ye, and B. Du, "Learn from others and be yourself in heterogeneous federated learning," in *CVPR*, 2022. **16, 17, 18**
- [171] W. Huang, M. Ye, Z. Shi, and B. Du, "Generalizable heterogeneous federated cross-correlation and instance similarity learning," *TPAMI*, 2023. **16, 17, 18**
- [172] Y. Luopan, R. Han, Q. Zhang, C. H. Liu, G. Wang, and L. Y. Chen, "Fedknow: Federated continual learning with signature task knowledge integration at edge," in *2023 IEEE 39th International Conference on Data Engineering (ICDE)*. IEEE, 2023, pp. 341–354. **16, 17, 18, 28, 29**
- [173] Y. Wu, B. Lorenzo, and B. Liu, "Lifelong learning for aoi and energy tradeoff optimization in satellite-airborne-terrestrial edge computing networks," in *2023 IEEE 34th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*. IEEE, 2023, pp. 1–7. **16, 17, 18**
- [174] X. Gao, X. Yang, H. Yu, Y. Kang, and T. Li, "Fedprok: Trustworthy federated class-incremental learning via prototypical feature knowledge transfer," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 4205–4214. **16, 17, 19**
- [175] D. Shenaj, M. Toldo, A. Rigon, and P. Zanuttigh, "Asynchronous federated continual learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 5055–5063. **16, 17, 19**
- [176] Y. Guo, T. Lin, and X. Tang, "Towards federated learning on time-evolving heterogeneous data," *arXiv preprint arXiv:2112.13246*, 2021. **16, 17**
- [177] L. Yuan, Y. Ma, L. Su, and Z. Wang, "Peer-to-peer federated continual learning for naturalistic driving action recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 5249–5258. **17, 28, 29**
- [178] M. Pennisi, F. P. Salanitri, G. Bellitto, C. Spampinato, S. Palazzo, B. Casella, and M. Aldinucci, "Experience replay as an effective strategy for optimizing decentralized federated learning," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 3376–3383. **17, 18**
- [179] T. Liao, L. Fu, J. Chen, W. Zhen, Z. Zheng, and C. Chen, "A swiss army knife for heterogeneous federated learning: Flexible coupling via trace norm," in *The Thirty-eighth Annual Conference on Neural Information Processing Systems*. **17, 18**
- [180] J. Dong, L. Wang, Z. Fang, G. Sun, S. Xu, X. Wang, and Q. Zhu, "Federated class-incremental learning," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 10 164–10 173. **17, 19, 26, 28, 30**
- [181] A. Montoya, D. Holman, S. data science, T. Smith, and W. Kan, "State farm distracted driver detection," 2016. [Online]. Available: <https://kaggle.com/competitions/state-farm-distracted-driver-detection> **17, 29**
- [182] N. Chondros, B. Zhang, T. Zacharias, P. Diamantopoulos, S. Maneas, C. Patsonakis, A. Delis, A. Kiayias, and M. Roussopoulos, "D-demos: A distributed, end-to-end verifiable, internet voting system," in *2016 IEEE 36th International Conference on Distributed Computing Systems (ICDCS)*. IEEE, 2016, pp. 711–720. **18**
- [183] Z. Zhong, J. Wang, W. Bao, J. Zhou, X. Zhu, and X. Zhang, "Semihfl: semi-supervised federated learning for heterogeneous devices," *Complex & Intelligent Systems*, vol. 9, no. 2, pp. 1995–2017, 2023. **18**
- [184] E. Trinh, "Tailoring attacks to federated continual learning models," Ph.D. dissertation, Delft University of Technology, 2023. **20, 21**
- [185] S. Wang, J. Hayase, G. Fanti, and S. Oh, "Towards a defense against federated backdoor attacks under continuous training," *arXiv preprint arXiv:2205.11736*, 2022. **20, 21, 22**
- [186] A. K. Chathoth, C. P. Necciai, A. Jagannatha, and S. Lee, "Differentially private federated continual learning with heterogeneous cohort privacy," in *2022 IEEE International Conference on Big Data (Big Data)*. IEEE, 2022, pp. 5682–5691. **20, 21**
- [187] Z. Han, C. Ge, B. Wu, and Z. Liu, "Lightweight privacy-preserving federated incremental decision trees," *IEEE Transactions on Services Computing*, 2022. **20, 21**
- [188] A. Chetouane and K. Karoui, "New continual federated learning system for intrusion detection in sdn-based edge computing," *Concurrency and*

- Computation: Practice and Experience*, vol. 37, no. 2, p. e8332, 2025. 20
- [189] C. Chen, P. Li, K. Sakurai *et al.*, “A multi-head federated continual learning approach for improved flexibility and robustness in edge environments,” *International Journal of Networking and Computing*, vol. 14, no. 2, pp. 123–144, 2024. 20
- [190] Y. Liu, X. Ma, J. Bailey, and F. Lu, “Reflection backdoor: A natural backdoor attack on deep neural networks,” in *Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part X 16*. Springer, 2020, pp. 182–199. 20, 21
- [191] J. Xiao, X. Tang, and S. Lu, “Privacy-preserving federated class-incremental learning,” *IEEE Transactions on Machine Learning in Communications and Networking*, 2023. 21
- [192] G. Zizzo, A. Rawat, N. Holohan, and S. Tirupathi, “Federated continual learning with differentially private data sharing,” in *Workshop on Federated Learning: Recent Advances and New Challenges (in Conjunction with NeurIPS 2022)*, 2022. 21
- [193] A. Talpur and M. Gurusamy, “Gfcl: A gru-based federated continual learning framework against data poisoning attacks in iot,” *arXiv preprint arXiv:2204.11010*, 2022. 21
- [194] X. Chen, C. Liu, B. Li, K. Lu, and D. Song, “Targeted backdoor attacks on deep learning systems using data poisoning,” *arXiv preprint arXiv:1712.05526*, 2017. 20
- [195] C. Dwork, “Differential privacy,” in *International colloquium on automata, languages, and programming*. Springer, 2006, pp. 1–12. 20
- [196] A. K. Chathoth, A. Jagannatha, and S. Lee, “Federated intrusion detection for iot with heterogeneous cohort privacy,” *arXiv preprint arXiv:2101.09878*, 2021. 20
- [197] C. Chen, Z. Lian, C. Su, and K. Sakurai, “Evaluating differential privacy in federated continual learning: A catastrophic forgetting-performance tradeoff analysis,” in *2024 Twelfth International Symposium on Computing and Networking (CANDAR)*. IEEE, 2024, pp. 135–141. 20
- [198] X. Chen, C. Luo, W. Wei, J. Xu, and S. Zhang, “Differential optimization federated incremental learning algorithm based on blockchain,” *Electronics*, vol. 11, no. 22, p. 3814, 2022. 20
- [199] R. Agrawal, J. Kiernan, R. Srikant, and Y. Xu, “Order preserving encryption for numeric data,” in *Proceedings of the 2004 ACM SIGMOD international conference on Management of data*, 2004, pp. 563–574. 21
- [200] V. Chandola, A. Banerjee, and V. Kumar, “Anomaly detection: A survey,” *ACM computing surveys (CSUR)*, vol. 41, no. 3, pp. 1–58, 2009. 21
- [201] K. Nogami, T. Hiroto, and G. Tanaka, “Federated learning with reservoir state analysis for time series anomaly detection,” *arXiv preprint arXiv:2502.05679*, 2025. 21
- [202] P. R. Tomas, P. Felix, L. Rosa, A. S. Gomes, and L. Cordeiro, “A novel approach for continual and federated network anomaly detection,” in *Proceedings of the Future Technologies Conference*. Springer, 2024, pp. 212–225. 21, 24
- [203] W. G. Halfond, J. Viegas, A. Orso *et al.*, “A classification of sql injection attacks and countermeasures,” in *ISSSE*, 2006. 21
- [204] M. Schreyer, H. Hemati, D. Borth, and M. A. Vasarhelyi, “Federated continual learning to detect accounting anomalies in financial auditing,” *arXiv preprint arXiv:2210.15051*, 2022. 21, 24
- [205] M. Beane, “Shadow learning: Building robotic surgical skill when approved means fail,” *Administrative Science Quarterly*, vol. 64, no. 1, pp. 87–123, 2019. 22
- [206] P. Kocher, J. Horn, A. Fogh, D. Genkin, D. Gruss, W. Haas, M. Hamburg, M. Lipp, S. Mangard, T. Prescher *et al.*, “Spectre attacks: Exploiting speculative execution,” *Communications of the ACM*, vol. 63, no. 7, pp. 93–101, 2020. 22
- [207] Z. Wang, F. Wu, F. Yu, Y. Zhou, J. Hu, and G. Min, “Federated continual learning for edge-ai: A comprehensive survey,” *arXiv preprint arXiv:2411.13740*, 2024. 22
- [208] A. Kumar, T. Braud, S. Tarkoma, and P. Hui, “Trustworthy ai in the age of pervasive computing and big data,” in *2020 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*. IEEE, 2020, pp. 1–6. 22
- [209] L. Guerdan and H. Gunes, “Federated continual learning for socially aware robotics,” in *2023 32nd IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*. IEEE, 2023, pp. 1522–1529. 22
- [210] J. Chen, J. He, J. Tang, W. Li, and Z. Yin, “Knowledge efficient federated continual learning for industrial edge systems,” *IEEE Transactions on Network Science and Engineering*, 2025. 22
- [211] S. Zhou, S. Shekhar, A. Chhokra, A. Dubey, and A. Gokhale, “Drift detection and adaptation for federated learning in iot with adaptive device management,” in *2024 IEEE International Conference on Big Data (BigData)*. IEEE, 2024, pp. 8088–8097. 22
- [212] J. Guerrero-Ibáñez, S. Zeadally, and J. Contreras-Castillo, “Sensor technologies for intelligent transportation systems,” *Sensors*, vol. 18, no. 4, p. 1212, 2018. 22
- [213] K. H. K. Reddy, R. S. Goswami, and D. S. Roy, “A deep learning-based smart service model for context-aware intelligent transportation system,” *The Journal of Supercomputing*, vol. 80, no. 4, pp. 4477–4499, 2024. 22, 24
- [214] L. Barbieri, S. Savazzi, M. Brambilla, and M. Nicoli, “Decentralized federated learning for extended sensing in 6g connected vehicles,” *Vehicular Communications*, vol. 33, p. 100396, 2022. 22, 24
- [215] R. Kapoor, J. Joshua, M. Vijayarangan *et al.*, “Fedcl-ensemble learning: A framework of federated continual learning with ensemble transfer learning enhanced for alzheimer’s mri classifications while preserving privacy,” *arXiv preprint arXiv:2411.12756*, 2024. 22
- [216] S. Hussain, I. Mubeen, N. Ullah, S. S. U. D. Shah, B. A. Khan, M. Zahoor, R. Ullah, F. A. Khan, and M. A. Sultan, “Modern diagnostic imaging technique applications and risk factors in the medical field: a review,” *BioMed research international*, vol. 2022, no. 1, p. 5164970, 2022. 22
- [217] K. Guo, T. Chen, S. Ren, N. Li, M. Hu, and J. Kang, “Federated learning empowered real-time medical data processing method for smart healthcare,” *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 2022. 22, 26
- [218] H. Sun, W. Zhang, R. Zhang, L. Xu, H. Guan, H. Zhao, B. Zhang, S. Yang, and X. Yu, “Federated continual learning based on weakly supervised diffusion models for disease diagnosis,” *IEEE Internet of Things Journal*, 2025. 22
- [219] Y. Xia, X. Liu, Y. Zhao, and Y. Wang, “Fcla-dt: Federated continual learning with authentication for distributed digital twin-based industrial iot,” *Journal of Communications and Information Networks*, vol. 9, no. 4, pp. 362–373, 2024. 23, 26
- [220] X. He, Q. Chen, L. Tang, W. Wang, T. Liu, L. Li, Q. Liu *et al.*, “Federated continuous learning based on stacked broad learning system assisted by digital twin networks: An incremental learning approach for intrusion detection in uav networks,” *IEEE Internet of Things Journal*, 2023. 23, 26, 27
- [221] J. Liu, Z. Dong, Z. Zhang, Z. Sun, and H. Ji, “Data sharing method of industrial internet of things based on federal incremental learning,” *Journal of Computer Applications*, vol. 42, no. 4, pp. 1235–1243, 2022. 23, 24
- [222] B. Liu, L. Wang, and M. Liu, “Lifelong federated reinforcement learning: a learning architecture for navigation in cloud robotic systems,” *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 4555–4562, 2019. 24
- [223] X. Yu, J. P. Queralta, and T. Westerlund, “Towards lifelong federated learning in autonomous mobile robots with continuous sim-to-real transfer,” *Procedia Computer Science*, vol. 210, pp. 86–93, 2022. 24
- [224] Z. Jin, J. Zhou, B. Li, X. Wu, and C. Duan, “Fl-iids: A novel federated learning-based incremental intrusion detection system,” 2024. 24
- [225] D. Jin, S. Chen, H. He, X. Jiang, S. Cheng, and J. Yang, “Federated incremental learning based evolvable intrusion detection system for zero-day attacks,” *Ieee Network*, vol. 37, no. 1, pp. 125–132, 2023. 24
- [226] M. A. P. Putra, N. W. Utami, I. G. J. E. Putra, N. Karna, A. Zainudin, and G. A. R. Sampedro, “Collaborative decentralized learning for detecting deepfake videos in entertainment,” in *2024 IEEE Gaming, Entertainment, and Media Conference (GEM)*. IEEE, 2024, pp. 1–4. 24
- [227] F. Lozano-Cuadra, B. Soret, I. Leyva-Mayorga, and P. Popovski, “Continual deep reinforcement learning for decentralized satellite routing,” *arXiv preprint arXiv:2405.12308*, 2024. 24
- [228] M. O. Osifeko, G. P. Hancke, and A. M. Abu-Mahfouz, “Surveilnet: A lightweight anomaly detection system for cooperative iot surveillance networks,” *IEEE Sensors Journal*, vol. 21, no. 22, pp. 25 293–25 306, 2021. 24
- [229] B. Alkouz, A. Bouguettaya, and A. Lakhdari, “Failure-sentient composition for swarm-based drone services,” in *2023 IEEE International Conference on Web Services (ICWS)*. IEEE, 2023, pp. 493–503. 24
- [230] L. Zhang, J. Zhu, D. Zhang, and Y. Liu, “An incremental photovoltaic power prediction method considering concept drift and privacy protection,” *Applied Energy*, vol. 351, p. 121919, 2023. 24
- [231] L. Zhang, J. Zhu, S. Li, A. Borghetti, and D. Zhang, “Online fault line detection in small-sample and streaming data environments,” *IEEE Transactions on Instrumentation and Measurement*, 2023. 24

- [232] Z. Lv, C. Cheng, and H. Lv, "Blockchain based decentralized learning for security in digital twins," *IEEE Internet of Things Journal*, 2023. [24](#)
- [233] Z. Guo, L. You, S. Liu, J. He, and B. Zuo, "Icmfed: An incremental and cost-efficient mechanism of federated meta-learning for driver distraction detection," *Mathematics*, vol. 11, no. 8, 2023. [23](#), [24](#)
- [234] C. Lanza, E. Angelats, M. Miozzo, and P. Dini, "Urban traffic forecasting using federated and continual learning," in *2023 6th Conference on Cloud and Internet of Things (CIoT)*. IEEE, 2023, pp. 1–8. [24](#)
- [235] Y. Lei, S. L. Wang, M. Zhong, M. Wang, and T. F. Ng, "A federated learning framework based on incremental weighting and diversity selection for internet of vehicles," *Electronics*, vol. 11, no. 22, p. 3668, 2022. [24](#), [25](#)
- [236] D. Neupane and J. Seok, "Bearing fault detection and diagnosis using case western reserve university dataset with deep learning approaches: A review," *Ieee Access*, vol. 8, pp. 93 155–93 178, 2020. [23](#)
- [237] P. Qi, X. Zhou, Y. Ding, S. Zheng, T. Jiang, and Z. Li, "Collaborative and incremental learning for modulation classification with heterogeneous local dataset in cognitive iot," *IEEE Transactions on Green Communications and Networking*, vol. 7, no. 2, pp. 881–893, 2023. [23](#)
- [238] M.-y. Zhu, Z. Chen, K.-f. Chen, N. Lv, and Y. Zhong, "Attention-based federated incremental learning for traffic classification in the internet of things," *Computer Communications*, vol. 185, pp. 168–175, 2022. [23](#)
- [239] A. W. Moore and D. Zuev, "Internet traffic classification using bayesian analysis techniques," in *Proceedings of the 2005 ACM SIGMETRICS international conference on Measurement and modeling of computer systems*, 2005, pp. 50–60. [23](#)
- [240] W. Wang, M. Zhu, J. Wang, X. Zeng, and Z. Yang, "End-to-end encrypted traffic classification with one-dimensional convolution neural networks," in *2017 IEEE international conference on intelligence and security informatics (ISI)*. IEEE, 2017, pp. 43–48. [23](#)
- [241] B. Cao, W. Wu, and J. Zhou, "Lcfl: A loss compensation mechanism for latest data in federated incremental learning," in *2022 IEEE 19th International Conference on Mobile Ad Hoc and Smart Systems (MASS)*, 2022, pp. 332–338. [23](#)
- [242] G. Zheng, M. A. Jacobs, V. Braverman, and V. S. Parekh, "Asynchronous decentralized federated lifelong learning for landmark localization in medical imaging," *arXiv preprint arXiv:2303.06783*, 2023. [25](#), [26](#)
- [243] H. Wang, R. He, X. Zhang, Z. Bian, D. Zeng, and J. Ma, "A peer-to-peer federated continual learning network for improving ct imaging from multiple institutions," *arXiv preprint arXiv:2306.02037*, 2023. [25](#), [26](#)
- [244] H. Zhang and G. Ren, "Intelligent leaf disease diagnosis: image algorithms using swin transformer and federated learning," *The Visual Computer*, pp. 1–24, 2024. [26](#)
- [245] J. Kim, M. H. Lim, K. Kim, and H.-J. Yoon, "Continual learning framework for a multicenter study with an application to electrocardiogram," *BMC Medical Informatics and Decision Making*, vol. 24, no. 1, p. 67, 2024. [26](#)
- [246] L. Sun, J. Wu, Y. Xu, and Y. Zhang, "A federated learning and blockchain framework for physiological signal classification based on continual learning," *Information Sciences*, vol. 630, pp. 586–598, 2023. [26](#)
- [247] F. Muhtasim Hossain, "Continual federated learning for network anomaly detection in 5g open-ran," Master's thesis, F. Muhtasim Hossain, 2023. [26](#)
- [248] C. Benzaid, F. M. Hossain, T. Taleb, P. M. Gómez, and M. Dieudonne, "A federated continual learning framework for sustainable network anomaly detection in o-ran," in *2024 IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, 2024, pp. 1–6. [26](#)
- [249] Y. Xia, Y. Chen, Y. Zhao, L. Kuang, X. Liu, J. Hu, and Z. Liu, "Fcllm-dt: Empowering federated continual learning with large language models for digital twin-based industrial iot," *IEEE Internet of Things Journal*, 2024. [26](#)
- [250] Y. Huang, C. Bert, S. Fischer, M. Schmidt, A. Dörfler, A. Maier, R. Fietkau, and F. Putz, "Continual learning for peer-to-peer federated learning: A study on automated brain metastasis identification. arxiv 2022," *arXiv preprint arXiv:2204.13591*. [25](#)
- [251] P. van Gelderen, X. Jiang, and J. H. Duyn, "Effects of magnetization transfer on t1 contrast in human brain white matter," *NeuroImage*, vol. 128, pp. 85–95, 2016. [25](#)
- [252] J. Yang, H. Veeraraghavan, S. G. Armato III, K. Farahani, J. S. Kirby, J. Kalpathy-Kramer, W. van Elmpt, A. Dekker, X. Han, X. Feng *et al.*, "Autosegmentation for thoracic radiation treatment planning: a grand challenge at aapm 2017," *Medical physics*, vol. 45, no. 10, pp. 4568–4581, 2018. [25](#)
- [253] K. Zhu, J. Xu, L. Zhou, X. Li, Y. Zhao, X. Xu, and S. Li, "Dmaf: data-model anti-forgetting for federated incremental learning," *Cluster Computing*, vol. 28, no. 1, p. 30, 2025. [28](#)
- [254] Y. Li, Y. Wang, T. Xiao, H. Wang, Y. Qi, and R. Li, "Rehearsal-free continual federated learning with synergistic regularization," 2024. [Online]. Available: <https://arxiv.org/abs/2412.13779> [27](#), [28](#)
- [255] Z. Yang, H. Wang, and Q. Sun, "Energy-efficient personalized federated continual learning on edge," *IEEE Embedded Systems Letters*, vol. 16, no. 4, pp. 345–348, 2024. [28](#), [29](#)
- [256] X. Zuo, Y. Luopan, R. Han, Q. Zhang, C. H. Liu, G. Wang, and L. Y. Chen, "Fedvit: Federated continual learning of vision transformer at edge," *Future Generation Computer Systems*, vol. 154, pp. 1–15, 2024. [27](#), [28](#), [29](#)
- [257] Y. Liu and D. Huang, "Sparse personalized federated class-incremental learning," *Information Sciences*, p. 121992, 2025. [27](#), [28](#), [30](#)
- [258] S. Hamdan, M. Ayyash, and S. Almajali, "Edge-computing architectures for internet of things applications: A survey," *Sensors*, vol. 20, no. 22, p. 6441, 2020. [27](#)
- [259] F. Javed, M. K. Afzal, M. Sharif, and B.-S. Kim, "Internet of things (iot) operating systems support, networking technologies, applications, and challenges: A comparative review," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 3, pp. 2062–2100, 2018. [27](#)
- [260] L. Wang, X. Zhang, H. Su, and J. Zhu, "A comprehensive survey of continual learning: theory, method and application," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024. [27](#)
- [261] S. Wang, D. C. Anastasiu, Z. Tang, M.-C. Chang, Y. Yao, L. Zheng, M. S. Rahman, M. S. Arya, A. Sharma, P. Chakraborty *et al.*, "The 8th ai city challenge," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 7261–7272. [29](#)
- [262] A. Krizhevsky, G. Hinton *et al.*, "Learning multiple layers of features from tiny images," 2009. [30](#)
- [263] Y. Le and X. Yang, "Tiny imagenet visual recognition challenge," *CS 231N*, vol. 7, no. 7, p. 3, 2015. [30](#)
- [264] Y. LeCun, C. Cortes, and C. Burges, "Mnist handwritten digit database," 2010. [30](#)
- [265] G. Cohen, S. Afshar, J. Tapson, and A. Van Schaik, "Emnist: Extending mnist to handwritten letters," in *2017 international joint conference on neural networks (IJCNN)*. IEEE, 2017, pp. 2921–2926. [30](#)
- [266] J. J. Hull, "A database for handwritten text recognition research," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 16, no. 5, pp. 550–554, 1994. [30](#)
- [267] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng, "Reading digits in natural images with unsupervised feature learning," 2011. [30](#)
- [268] K. Saenko, B. Kulis, M. Fritz, and T. Darrell, "Adapting visual category models to new domains," in *Computer Vision—ECCV 2010: 11th European Conference on Computer Vision, Heraklion, Crete, Greece, September 5–11, 2010, Proceedings, Part IV 11*. Springer, 2010, pp. 213–226. [30](#)
- [269] Y. Zhang and B. D. Davison, "Impact of imagenet model selection on domain adaptation," 2020. [30](#)
- [270] Y. Ma, Z. Xie, J. Wang, K. Chen, and L. Shou, "Continual federated learning based on knowledge distillation," in *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, L. D. Raedt, Ed. International Joint Conferences on Artificial Intelligence Organization, 7 2022, pp. 2182–2188, main Track. [Online]. Available: <https://doi.org/10.24963/ijcai.2022/303> [30](#)
- [271] T. Minka, "Estimating a dirichlet distribution," 2000. [30](#)
- [272] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778. [31](#)
- [273] Y. Li, Y. Wang, J. Dong, H. Wang, Y. Qi, R. Zhang, and R. Li, "Resource-constrained federated continual learning: What does matter?" 2025. [Online]. Available: <https://arxiv.org/abs/2501.08737> [32](#)
- [274] S. Woo, S. Debnath, R. Hu, X. Chen, Z. Liu, I. S. Kweon, and S. Xie, "Convnext v2: Co-designing and scaling convnets with masked autoencoders," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 16 133–16 142. [32](#)
- [275] W. Huang, J. Liang, Z. Shi, D. Zhu, G. Wan, H. Li, B. Du, D. Tao, and M. Ye, "Learn from downstream and be yourself in multimodal large language model fine-tuning," 2024. [Online]. Available: <https://arxiv.org/abs/2411.10928> [32](#)

- [276] Y. Yang, J. Zhou, X. Ding, T. Huai, S. Liu, Q. Chen, Y. Xie, and L. He, "Recent advances of foundation language models-based continual learning: A survey," *ACM Computing Surveys*, vol. 57, no. 5, pp. 1–38, 2025. [32](#)
- [277] W. Zhao, S. Wang, Y. Hu, Y. Zhao, B. Qin, X. Zhang, Q. Yang, D. Xu, and W. Che, "Sapt: A shared attention framework for parameter-efficient continual learning of large language models," *arXiv preprint arXiv:2401.08295*, 2024. [32](#)
- [278] D. Jannach, A. Manzoor, W. Cai, and L. Chen, "A survey on conversational recommender systems," *ACM Computing Surveys (CSUR)*, vol. 54, no. 5, pp. 1–36, 2021. [33](#)
- [279] S. K. Dam, C. S. Hong, Y. Qiao, and C. Zhang, "A complete survey on llm-based ai chatbots," *arXiv preprint arXiv:2406.16937*, 2024. [33](#)
- [280] C. Zhang, C. Zhang, M. Zhang, and I. S. Kweon, "Text-to-image diffusion models in generative ai: A survey," *arXiv preprint arXiv:2303.07909*, 2023. [33](#)



**Yichen Li** is currently a Ph.D. candidate in the School of Computer Science and Technology at Huazhong University of Science and Technology. Prior to this, he obtained his bachelor's degree from the School of Computer Science and Technology at Soochow University. His research interests include Federated Learning, Recommendation Systems, and Software Engineering. As the first author, he has published several papers at top-tier machine learning and computer vision conferences and journals such as CVPR, ECCV, ICML, ICLR, and IEEE TPAMI.



**Haozhao Wang** is currently an assistant professor in the School of Computer Science and Technology at Huazhong University of Science and Technology. Before that, he worked as a Postdoc and received a Ph.D. degree at the same department and school. He received the B.S. degree in computer science at the University of Electronic Science and Technology in 2016. He also worked as a research fellow in SLab at Nanyang Technical University. His research interests include Distributed Machine Learning and Multi-modal Learning.



His research interests includes wireless communication, Internet of things, distributed computing and AI enabled networking.



**Tianzhe Xiao** received the B.E. degree from Wuhan University of Science and Technology in 2021. He is currently working toward his PhD degree with the School of Computer Science and Technology, Huazhong University of Science and Technology. His research interests include federated learning and machine learning security.



**Hong Liu** received her B.E. degree from Huazhong University of Science and Technology in 2022. She is currently working toward her master's degree in School of Computer Science and Technology, Huazhong University of Science and Technology. Her research interests include federated learning and continual learning.



**Minzhu Tu** is currently working toward her B.E. degree with the School of Computer Science and Technology, Beijing University of Posts and Telecommunications. Her research interests include Federated Learning and Software Engineering.



**Yuying Wang** received the B.E. degree from Shandong Normal University in 2023. She is currently working toward her master's degree with the School of Computer Science and Technology, Soochow University. Her research interests include Federated Learning and Software Engineering.



**Xin Yang** received an M.S. degree in electronic engineering from Sichuan University, China, in 2010, and the Ph.D. degree in computer science from Southwest Jiaotong University, Chengdu, in 2019. He is currently a Professor at the School of Computing and Artificial Intelligence, Southwestern University of Finance and Economics. He has authored more than 70 research papers in refereed journals and conferences. His research interests include machine learning, continual learning and multi-granularity learning.



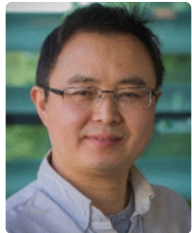
**Rui Zhang** is a professor in the School of Computer Science and Technology at Huazhong University of Science and Technology. He is a Visiting Professor at Tsinghua University and was a Professor at the School of Computing and Information Systems of the University of Melbourne. He has won several awards including the prestigious Future Fellowship by the Australian Research Council in 2012, Chris Wallace Award for Outstanding Research by the Computing Research and Education Association of Australasia (CORE) in 2015, and Google Faculty

Research Award in 2017. His inventions have been adopted by major IT companies such as Microsoft, Amazon and AT&T. He obtained his Bachelor's degree from Tsinghua University in 2001, PhD from National University of Singapore in 2006, and has then started as a faculty member in The University of Melbourne since 2007. Before joining the University of Melbourne, he has been a visiting research scientist at AT&T labs-research in New Jersey and at Microsoft Research in Redmond, Washington. He has also been a regular visiting researcher at Microsoft Research Asia in Beijing. His research interests include big data and AI, particularly in areas of recommendation systems, knowledge bases, chatbot, spatial and temporal data analytics, moving object management and data streams.



**Ruixuan Li** is a professor in the School of Computer Science and Technology at Huazhong University of Science and Technology. He received the B.S., M.S. and Ph.D. in computer science from Huazhong University of Science and Technology, China in 1997, 2000 and 2004 respectively. He was the visiting scholar of the University of Western Sydney (UWS), Australia, in 2005. He was the visiting researcher of the University of Toronto (UofT), Canada, from June 2009 to October 2010. He also holds an Adjunct Professor position at Concordia University, an Adjunct

Senior Fellow position at University of Western Sydney (UWS), and a Guest Researcher position at Institute of Information Engineering, Chinese Academy of Sciences. His research interests include cloud and edge computing, big data management, and distributed system security. He is a member of IEEE and ACM; is a distinguished member of China Computer Federation (CCF); is the Vice Director of Technical Committee of Distributed Computing and System of CCF, and Technical Committee of Information System of CCF.



**Shui Yu** is Professor of the School of Computer Science in the Faculty of Engineering and Information Technology at UTS, the Deputy Chair of the UTS Research Committee, and is a researcher of cybersecurity, privacy and the networking, communication aspects of Big Data, and applied mathematics for computer science. Many of his research outputs have been adopted by industry, for example, the auto scale strategy of Amazon Cloud against distributed denial-of-service attacks. Shui serves his research communities in various roles, including serving on

the editorial boards of IEEE Communications Surveys and Tutorials, IEEE Communications Magazine and the IEEE Internet of Things Journal, among others. He has been a member of organizing committees for many international conferences, such as the publication chair for IEEE Globecom 2015, IEEE INFOCOM 2016 and 2017, TPC chair for IEEE Big Data Service 2015, and general chair for ACSW 2017. He served as a Distinguished Lecturer of IEEE Communications Society (2018-2021). He is a Distinguished Visitor of IEEE Computer Society (2022-2024), a voting member of IEEE ComSoc Educational Services board, and an elected member of Board of Governors of IEEE Communications Society and IEEE Vehicular Technology Society, respectively. He is a Fellow of IEEE.



**Song Guo** is a chair professor in the Department of Computer Science and Engineering at the Hong Kong University of Science and Technology. Before joining HKUST in 2023, he was a Professor at The Hong Kong Polytechnic University. He also holds a Changjiang Chair Professorship awarded by the Ministry of Education of China. Prof. Guo is a Fellow of the Canadian Academy of Engineering and a Fellow of the IEEE (Computer Society). His research interests are mainly in big data, edge AI, mobile computing, and distributed systems. He

published many papers in top venues with wide impact in these areas and was recognized as a Highly Cited Researcher (Clarivate Web of Science). He is the recipient of over a dozen Best Paper Awards from IEEE/ACM conferences, journals, and technical committees. Prof. Guo is the Editor-in-Chief of IEEE Open Journal of the Computer Society and the Chair of IEEE Communications Society (ComSoc) Space and Satellite Communications Technical Committee. He was an IEEE ComSoc Distinguished Lecturer and a member of IEEE ComSoc Board of Governors. He has served the IEEE Computer Society on the Fellow Evaluation Committee and has been named on the editorial board of a number of prestigious international journals like IEEE TPDS, IEEE TCC, IEEE TETC, etc. He has also served as chair of organizing and technical committees of many international conferences.