FedProK: Trustworthy Federated Class-Incremental Learning via Prototypical Feature Knowledge Transfer

Xin Gao[1]* Xin Yang[1]† Hao Yu[1]* Yan Kang[2], Tianrui Li[3]
[1]Southwestern University of Finance and Economics, [2]Webank, [3]Southwest Jiaotong University
xingaocs@foxmail.com, yangxin@swufe.edu.cn,
yuhao2033@163.com, yangkang@webank.com, trli@swjtu.edu.cn

Abstract

*Federated Class-Incremental Learning (FCIL) focuses on continually transferring the previous knowledge to learn new classes in dynamic Federated Learning (FL). However, existing methods do not consider the trustworthiness of FCIL, i.e., improving continual utility, privacy, and efficiency simultaneously, which is greatly influenced by catastrophic forgetting and data heterogeneity among clients. To address this issue, we propose FedProK (**Fed**erated **Pro**totypical Feature **K**nowledge Transfer), leveraging prototypical feature as a novel representation of knowledge to perform spatial-temporal knowledge transfer. Specifically, FedProK consists of two components: (1) feature translation procedure on the client side by temporal knowledge transfer from the learned classes and (2) prototypical knowledge fusion on the server side by spatial knowledge transfer among clients. Extensive experiments conducted in both synchronous and asynchronous settings demonstrate that our FedProK outperforms the other state-of-the-art methods in three perspectives of trustworthiness, validating its effectiveness in selectively transferring spatial-temporal knowledge.*

1. Introduction

Artificial Intelligence (AI) has profoundly impacted humanity in various aspects [9, 33]. However, with the increasing awareness of ethics, fairness, and security, the centralized training and data storage paradigm in traditional deep learning has been greatly challenged [1, 28]. The academic and industrial communities call for the establishment of trustworthy AI systems.

Federated Class Incremental Learning (FCIL) is a dynamic and heterogeneous paradigm of Federated Learning (FL) [22, 27], depicting a more realistic scenario of iterative federated training: data reaches clients sequentially in the form of task streams. While its privacy preservation and adaptivity to non-stationary environments make FCIL a promising solution for trustworthy AI, it is faced with two key challenges. The first challenge is the **catastrophic forgetting** of the previous knowledge [39]. After completing the training on new tasks, models tend to forget the knowledge of previous tasks due to the overwriting of parameters [8, 41, 43]. Another key challenge is **data heterogeneity** among clients, which causes severe performance degradation and unstable convergence of the global model [25, 28, 42]. By introducing diverse and varying data distributions across clients, data heterogeneity significantly exacerbates catastrophic forgetting [4]. As illustrated in Fig. 1, the intertwined challenges of FCIL require more effective approaches to knowledge transfer.

However, existing FCIL methods exhibit certain limitations in addressing these issues. A typical approach of FCIL is to store or generate exemplars of previous classes and replay them when training on new data [6, 8, 43]. How-

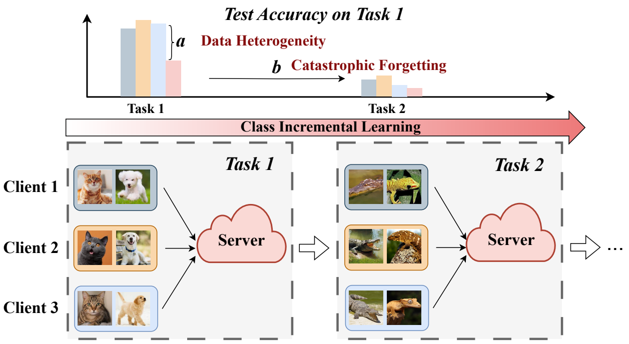


Figure 1. Illustration of the catastrophic forgetting and data heterogeneity in FCIL. The bar chart shows the test accuracy when vanilla FedAvg is employed in this scenario. a: The heterogeneous local data leads to poor performance of the global model. b: After the completion of training on Task 2, the model suffers catastrophic forgetting of Task 1.

*Equal contribution
†Corresponding author

ever, the performance is heavily influenced by the size of the exemplar set, and the continually growing number of exemplars may exceed the memory capacity of edge devices [8]. [43] generates pseudo samples on the server and distributes them to the clients, which greatly increases the communication cost of federated training and vulnerability of potential attacks. Former works on traditional centralized continual learning (CL) emphasized that a capable CL model should avoid overfitting to previous tasks when combating catastrophic forgetting [17, 37, 47]. Instead, a capable CL model should strike a delicate balance: preserving previous knowledge (stability) while accommodating new knowledge (plasticity), which is completely overlooked in current FCIL research [39]. Besides, simply integrating continual learning methods into the federated setting demonstrates poor performance [26], which is tested in our experiments.

Therefore, we formulate a trustworthy FCIL framework to evaluate the effectiveness of spatial-temporal knowledge transfer, which comprises three main aspects: **continual utility**, **efficiency**, and **privacy**. Continual utility represents the performance of the global model on overcoming forgetting and data heterogeneity, which can be tested with the final accuracy of the global model and the trade-off of the stability-plasticity dilemma. To enhance privacy preservation, FCIL algorithms should avoid data leakage in addressing catastrophic forgetting. As for efficiency, communicational cost and computational cost should be measured and controlled to an acceptable level.

In order to achieve spatial-temporal knowledge transfer in FCIL under a trustworthy framework, we propose FedProK (**Fed**erated **Pro**totypical Feature **K**nowledge Transfer). FedProK consists of two components: (1) **feature translation** procedure on the client side that achieves temporal knowledge transfer among tasks to mitigate catastrophic forgetting, and (2) **prototypical knowledge fusion** on the server side that enables spatial knowledge transfer across clients to address data heterogeneity.

The contributions of this paper are as follows:

1. We raise the issue of trustworthiness in FCIL and formulate a comprehensive framework of trustworthy FCIL concerning continual utility, efficiency, and privacy.
2. We propose FedProK to achieve temporal knowledge transfer among tasks and spatial knowledge transfer across clients under the trustworthy FCIL framework.
3. We evaluate the effectiveness of our method in knowledge transfer with extensive experiments, including various settings of class increments and data heterogeneity.

2. Related Work

2.1. Federated Class-Incremental Learning

Class Incremental Learning (CIL) methods aim to overcome catastrophic forgetting of the previous classes when training on the new classes [18, 29, 30, 47]. With the growing awareness of data privacy, FCIL has emerged as a novel approach of dynamic, heterogeneous and distributed machine learning paradigm [39].

FCIL can be categorized into synchronous and asynchronous based on whether the task orders are the same among clients [39]. Early efforts often directly integrate existing CL methods in FL. FedCurv [32] adds a regularization term in the loss function to penalize the changes in important parameters of previous tasks. GradMA[24] utilizes quadratic programming to correct the direction of local updates. [26] performs knowledge distillation on both the client and the server side according to a surrogate dataset. Recently, studies have started to focus on more complex scenarios and methods. FedSW [48] jointly optimize the global model and local models with self-paced adaptive sample weights. FedSSD [11] transfers the knowledge in the previous global model to the local updates in the current round with a selective knowledge distillation mechanism. Besides knowledge distillation and regularization, experience replay is also commonly employed in FCIL [39]. FedWeIT [8] constructs an exemplar memory to retain the samples from the previous classes and performs rehearsal when training on new tasks. TARGET [43] replays globally synthesized pseudo data on the clients to mitigate forgetting and alleviate different degrees of heterogeneity.

Generally, current FCIL methods predominantly focus on overcoming catastrophic forgetting, ignoring efficiency and privacy. Some methods even store and replay a shared core set of previous classes [36, 43], severely compromising the privacy of FL. Besides, the exploration of stability and plasticity dilemma in FCIL remains unaddressed [5, 41].

2.2. Trustworthy Federated Learning

FL is naturally more aligned with the core of trustworthy AI than the centralized training paradigm [3, 14]. To enhance its trustworthiness, researchers delve into the privacy, efficiency, fairness, and interpretability of FL. [40] designs a blockchain-based FL framework to ensure systematic security and privacy. [23] focuses on fairness and accountability of blockchain-based FL to improve the trustworthiness of medical diagnoses. [49] adopts the elitist non-dominated sorting genetic algorithm (NSGA-II) to minimize the communication overhead and maximize the model accuracy. [13] utilizes scalable encoding methods and real-time federated evolutionary strategies to optimize neural network models for reducing communication costs and improving performance on edge devices.

A trustworthy FL system typically involves more than one objective [10, 15] to optimize. [46] demonstrates that simultaneously optimizing utility, privacy, and efficiency to its optimal is impossible. For instance, differential privacy adds Gaussian or Laplace noise to obfuscate the network parameters transmitting between clients and the server, but excessive noise can severely degrade the accuracy of the model [44]. Similarly, while homomorphic encryption safeguards data privacy by translating parameters into ciphertext, overly sophisticated encryption and decryption processes may reduce model efficiency [45].

Current works on trustworthy FL mainly focus on stationary data and there is no trustworthy FCIL framework yet [2, 21]. However, in the real-world scenario, a dynamic and heterogeneous FCIL setting is worth investigating.

3. Problem Definition

3.1. Federated Class-Incremental Learning

As the combination of FL and CIL, there is a class-incremental task sequence $\mathcal{T}_k = \{\mathcal{T}_k^1, \ldots, \mathcal{T}_k^T\}$ on each local client k in FCIL, where T is the total number of incremental states. The composition of classes in the local task sequences can either be the same across clients (i.e., synchronous FCIL) or different across clients (i.e., asynchronous FCIL). Not every communication round $r \in R$ involves class increments [7]. Communication rounds alternate between incremental state and interval rounds, and the dataset $\mathcal{D}^t$ and its class set $\mathcal{C}^t$ of the interval rounds are consistent with those of the latest incremental state. We denote the local task at t-th increments and its subsequent interval rounds on client k as $\mathcal{T}_k^t = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{N^t}$, where N^t denotes the number of samples of task t. Thus, the entire label space at t-th task is $\mathcal{Y}^t = \cup_{k=1}^{K} \mathcal{Y}_k^t$, which includes:

- New classes: $\mathcal{C}^t = \cup_{k=1}^{K} \mathcal{C}_k^t$,
- Previous classes: $\cup_{i=1}^{t-1} \mathcal{C}^i$.

The standard objective of FCIL is to train a global model Θ_G that is capable of discriminating samples within the entire label space $\mathcal{Y}^T$, i.e., the superset of classes observed by all of the clients in the federated system. It is mainly challenged by catastrophic forgetting and data heterogeneity issues. Catastrophic forgetting refers to the severe degradation of model performance in previous classes. Data heterogeneity among clients incurs the deviation between the individual optima of the local models and the true optimum of the global model [20]. In FCIL, particularly, data heterogeneity exacerbates catastrophic forgetting by overwriting the important weights of previous classes more severely [39]. In this paper, we explore the intertwined challenge of class-incremental learning among tasks and data heterogeneity among clients.

3.2. Synchronous and Asynchronous Federated Class-Incremental Learning

FCIL can be categorized into synchronous FCIL and asynchronous FCIL according to the different forms of data heterogeneity [39].

Synchronous FCIL. In this scenario, the task $\mathcal{T}_k$ at the same incremental state t contains the data of identical classes $\mathcal{C}_k^t = \mathcal{C}_K^t$ but varied amounts of samples for each class. It is caused by distribution-based label imbalance among clients [20]. Each client is allocated a proportion of the samples of each label according to Dirichlet distribution. The degree of heterogeneity is controlled with the concentration parameter of Dirichlet distribution α. A smaller value of α results in a more heterogeneous data partition.

Asynchronous FCIL. In this scenario, the task $\mathcal{T}_k$ at the same incremental state t contains the data of different classes $\mathcal{C}_k^t \neq \mathcal{C}_l^t$ $(k, l \in K)$. It is caused by quantity-based label imbalance among clients [20]. We refer to the proportion of overlapping classes in the total number of classes $\cup_{t=1}^{T} \mathcal{C}_k^t \cap \cup_{t=1}^{T} \mathcal{C}_l^t$ $(k, l \in K)$ in the local task sequence $\mathcal{T}_k, \mathcal{T}_l$ as consensus rate γ. For instance, if each client's local task sequence comprises samples of 50 classes, with 25 common classes and 25 unique classes, the consensus rate γ in this case is 0.5. Intuitively, a lower γ in the federated system indicates a higher degree of heterogeneity.

Notably, in both cases, we followed the previous works [8, 35, 41] to assume that the numbers of clients K are consistent during the entire training.

4. Trustworthy Framework of Federated Class-Incremental Learning

By achieving spatial-temporal knowledge transfer among tasks and across clients, the FCIL global model gains the capability of discriminating the superset of classes observed by all clients in the federated system. There has been plenty of work dedicated to designing such a model. However, existing works exhibit limitations in the following aspects:

- Overemphasizing the preservation of previous knowledge (stability) while ignoring the accommodation of new knowledge (plasticity).
- A complicated model design may increase the computational cost of local update and aggregation, and a massive number of parameters transmitted between clients and the server may increase the communication cost.
- Some approaches to mitigate forgetting and data heterogeneity may break the privacy protocol of FL, especially exemplar-based approaches.

A trustworthy FL algorithm must satisfy multiple conflicting objectives simultaneously, which includes three major objectives, i.e., privacy, efficiency, and utility, and several minor objectives, i.e., robustness, interpretability, fairness, etc. [12, 16]. Therefore, we propose a trustworthy framework of FCIL to consolidate multiple objectives:

- **Continual Utility**: trustworthy FCIL models must handle the classic **plasticity-stability** dilemma inherent in CL with more effective spatial-temporal knowledge transfer.
- **Privacy**: as a main driving force of FL, **data privacy** should not be neglected. In this way, a shared buffer of private data must be avoided.
- **Efficiency**: mitigating forgetting requires extra training costs, like knowledge distillation and generative replay. **Computational cost and communication cost** should be measured and controlled to an acceptable range.

Utility refers to the model's performance, typically assessed through the average accuracy [4, 38], which is not sufficient for FCIL. To find trade-offs between the stability-plasticity dilemma of continual learning, we evaluate stability and plasticity, respectively, with averaged accuracy on the test set of previous classes and current classes during the entire training. After the completion of training on task t, the test accuracy on the previous tasks $\mathcal{A}_G^{t-1}$ and on the current task $\mathcal{A}_G^t$ reflect the stability and plasticity of the global model θ_G. Note that the test set to produce $\mathcal{A}_G^{t-1}$ contains samples of $\cup_{i=1}^{t-1}\mathcal{C}^i$ (the classes of previous tasks) and the test set to produce $\mathcal{A}_G^t$ contains samples of $\mathcal{C}^t$ (the classes of the current task). Based on this, we formulate the following equation to assess continual utility, i.e. the trade-off between stability and plasticity in FCIL:

$$U(\theta_G) = \lambda \times \mathcal{A}_G^{t-1} + (1-\lambda) \times \mathcal{A}_G^t, \tag{1}$$

where λ is the hyperparameter and θ_G is the parameters of the global model G.

Privacy refers to the effectiveness of avoiding the leakage of the original information [8]. We assess privacy by measuring the similarity between the reconstructed images generated by malicious attackers and the original local images of clients, which is defined as follows:

$$P(\theta_G) = 1 - \frac{1}{1 + \text{MSE}(img_{true}, img_{pred})}, \tag{2}$$

where MSE is the mean square error, img_{true} and img_{pred} denote the numpy arrays of the ground truth image and the reconstructed image.

Efficiency refers to the model's total costs of communication and computation. We assess the efficiency of methods with the time to perform one epoch of federated iteration [35], which is defined as:

$$E(\theta_G) = \frac{1}{R}(\tau_1 + \tau_2), \tag{3}$$

where R denotes the number of rounds, τ_1 and τ_2 denote the communication cost and computation cost. τ_1 reflects the overhead of data transmission between server and clients, and τ_2 reflects the total overhead of local updates and model aggregation.

Algorithm 1 FedProK

Input: local dataset $\{D_1, \dots, D_K\}$, global model G with initial weights θ_G^0, number of local epochs E, number of communication rounds R, number of incremental tasks T

Output: global model G with final weights θ_G^R

```
for $r = 1$ to $R$ do
  for each client $k$ do
    Compute task number $t = r//(R/T) + 1$
    Compute prototype $\mu_{k,c}^t$ for each new class $c$
    if $t > 1$ then
      Compute pair-wise relations of previous classes and new classes, select the base class $n \in \mathcal{C}^t$ of feature translation for each previous class $p \in \cup_{i=1}^{t-1}\mathcal{C}^i$: $n = \arg\min_n \text{dis}(\mu_{k,p}^t, \mu_{k,c}^t)$
      Perform feature translation for each previous class $\hat{f}^t(\mathbf{x}_p) = f(\mathbf{x}_n) + \mu_{k,p}^t - \mu_{k,n}^t$
    end if
    Local training on current task $\mathcal{T}_k^t$ with pseudo feature $\hat{f}^t(\mathbf{x}_p)$
    Upload local weights $\theta_k^r$
    Transfer local prototypical knowledge $\mathcal{P}_k^r$
  end for
  Aggregate model weights with standard FedAvg $\frac{1}{K}\sum_{k=1}^{K}\theta_k^r$
  Construct the prototypical knowledge base $\mathcal{P}_G^t$
  Distribute weights of global model $\theta_G^r$ to clients
  Transfer global prototypical knowledge $\mathcal{L}_G^r$ to clients
end for
```

5. The Proposed Method

The overview of FedProK is depicted in Fig. 2 and summarized in Algorithm 1. FedProK comprises two novel components: a client-side feature translation procedure to achieve temporal knowledge transfer and a server-side prototypical knowledge fusion mechanism to achieve spatial knowledge transfer. During the local update phase, clients train on their private data, compute the prototype for the new classes, and perform feature translation to mitigate forgetting of the previous classes. The prototype lists, as well as the weights of local networks, are uploaded to the server. During the global aggregation phase, the server needs to perform a standard FedAvg to aggregate the model parameters and a prototypical aggregation to construct a global knowledge base. Then, the class-wise prototype list is distributed back to the clients. Since the knowledge base contains the global prototypical list of previous classes without reserving or globally generating any exemplars, FedProK can effectively transfer knowledge across clients and among tasks without compromising data privacy.

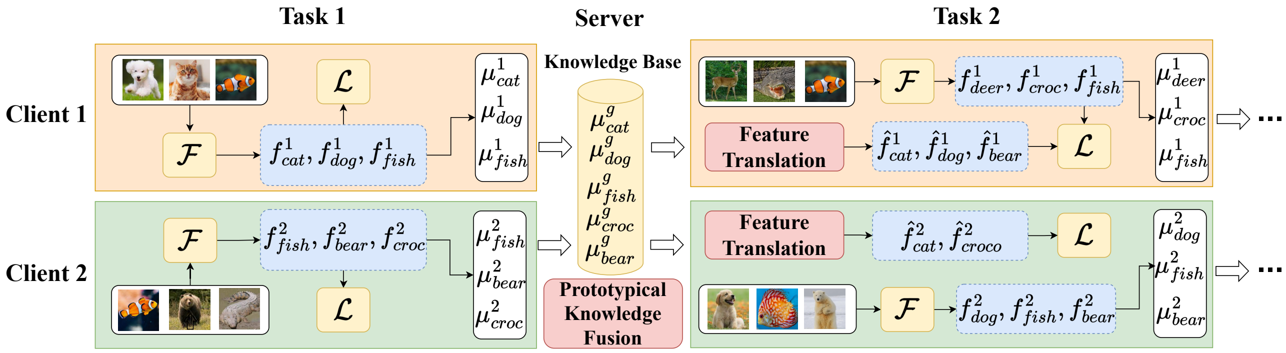

Task 1
Server
Task 2
Knowledge Base
Client 1
Client 2
$\mathcal{L}$
$\mathcal{F}$
$f^1_{cat}, f^1_{dog}, f^1_{fish}$
μ^1_{cat} μ^1_{dog} μ^1_{fish}
$f^2_{fish}, f^2_{bear}, f^2_{croc}$
μ^2_{fish} μ^2_{bear} μ^2_{croc}
μ^g_{cat} μ^g_{dog} μ^g_{fish} μ^g_{croc} μ^g_{bear}
Prototypical Knowledge Fusion
$f^1_{deer}, f^1_{croc}, f^1_{fish}$
Feature Translation
$\hat{f}^1_{cat}, \hat{f}^1_{dog}, \hat{f}^1_{bear}$
μ^1_{deer} μ^1_{croc} μ^1_{fish}
$\hat{f}^2_{cat}, \hat{f}^2_{croco}$
$f^2_{dog}, f^2_{fish}, f^2_{bear}$
μ^2_{dog} μ^2_{fish} μ^2_{bear}
...

Figure 2. Illustration of the knowledge transfer via FedProK. The clients conduct feature translation locally to mitigate forgetting of the previous classes of the entire federated system. The server conducts prototypical knowledge fusion to alleviate the adverse impact of data heterogeneity among clients. $\mathcal{F}$ and $\mathcal{L}$ denote the feature extractor and the classification layer.

5.1. Feature Translation

To alleviate the catastrophic forgetting of past knowledge, we utilize a feature translation mechanism locally to enable temporal knowledge transfer among tasks. Feature translation module is a combination of a real feature extractor $\mathcal{F}$ and a pseudo-feature generator $\mathcal{G}$. Firstly, $\mathcal{F}$ extracts the feature vectors of sample $(\mathbf{x}_i, \mathbf{y}_i)$:

$$f(\mathbf{x}_i) = \mathcal{F}(\mathbf{x}_i, \mathbf{y}_i). \tag{4}$$

After that, client k computes the local prototype μ for each class and reserves it in the local prototypical list $\mathcal{L}^t_k$:

$$\mu^t_{k,c} = \frac{1}{|D_{k,c}|}\sum_{i=1}^{|D_{k,c}|} f(\mathbf{x}_i), \tag{5}$$

where $|D_{k,c}|$ denotes the number of local samples belonging to class c on client k. When new data arrives, $\mathcal{F}$ is frozen, and the generator $\mathcal{G}$ generates pseudo feature vectors of previous classes according to the real feature of a selected new class and the prototype of the past classes.

Generally, each incremental state may involve more than one new class, there are several strategies for selecting the base class of feature translation. The simplest strategy is to randomly select one from all new classes as the base class. However, we observed that the random selection performs poorly when the inter-class relation is relatively large. Therefore, for a previous class $p \in \cup^{t-1}_{i=1}\mathcal{C}^i$ and a new class $c \in \mathcal{C}^t$, we compute the pair-wise relation with cosine similarity:

$$\text{dis}(\mu^t_{k,p}, \mu^t_{k,c}) = \frac{\mu^t_{k,p} \cdot \mu^t_{k,c}}{\left\|\mu^t_{k,p}\right\|_2 \times \left\|\mu^t_{k,c}\right\|_2}. \tag{6}$$

And we select the nearest new class n as the geometric base of feature translation for class p to simulate the representation of the previous class p to the greatest extent:

$$n = \arg\min_c \text{dis}(\mu^t_{k,p}, \mu^t_{k,c}). \tag{7}$$

Through the translation process in Eq. (8), the pseudo feature vector is generated by translating the value of each dimension with the distance between the values of the corresponding dimension of $\mu^t_{k,n}$ and $\mu^t_{k,p}$. Rehearse them by directly sending them to the classification layer alongside new inputs.

Since $\mathcal{F}$ is all layers of the backbone except for the classification layer, the straightforward feature translation process is much more computationally efficient than other generative replay methods [29], i.e., Generative Adversarial Net (GAN) and Variational Auto-Encoder (VAE). At task t, the feature translation from a certain new class $n \in \mathcal{C}^t$ to previous class $p \cup^{t-1}_{i=1} \mathcal{C}^i$ is defined as:

$$\hat{f}^t(\mathbf{x}_p) = f(\mathbf{x}_n) + \mu^t_{k,p} - \mu^t_{k,n}, \tag{8}$$

where $f(\mathbf{x}_n)$ denotes the feature of a real sample $\mathbf{x}_n$ in class n extracted by $\mathcal{F}$, and $\hat{f}^t(\mathbf{x}_p)$ denotes the pseudo feature vector of a pseudo sample $\mathbf{x}_p$ of class p produced in the t-th incremental state by $\mathcal{G}$. Notably, $\mu^t_{k,n}$ is currently computed while $\mu^t_{k,p}$ is previously stored in the $\mathcal{P}^t_k$.

5.2. Prototypical Knowledge Fusion

To address data heterogeneity, we construct a knowledge base on the server with prototypical knowledge fusion mechanism to achieve spatial knowledge transfer among clients. At each round r with its corresponding task t, client k computes the local prototype for each class $c \in \mathcal{C}^t_k$ and uploads the local prototypical list $\mathcal{P}_k$ to the server.

In synchronous FCIL and asynchronous FCIL with a consensus rate γ greater than zero, local class sets overlap across clients, which is defined as $\mathcal{C}^t_i \cap \mathcal{C}^t_j \neq \emptyset(i, j \in K)$. In this case, the uploaded local prototypical lists $\mathcal{P}_i$ and $\mathcal{P}_j$ might contain different prototypes $\mu^t_{i,c} \neq \mu^t_{j,c}$ for the same class c. Besides, if the current dataset includes previous classes, the knowledge base $\mathcal{P}^t_G$ on the server should be updated to rectify the bias of previous data and alleviate the impacts of concept drift [7]. Therefore, we design the prototypical knowledge fusion mechanism that horizontally aggregates heterogeneous prototypical knowledge from different clients and vertically fuses previous and new knowledge along the timeline. The fusion process is defined as:

$$\mu_c^t = \begin{cases} \sum_{k=1}^{K} \frac{|D_{k,c}|}{|D_c|} \mu_{k,c}^r, & c \notin \cup_{i=1}^{t-1} \mathcal{C}^i \\ \beta(\sum_{k=1}^{K} \frac{|D_{k,c}|}{|D_c|} \mu_{k,c}^r) + (1-\beta)\mu_c^{t-1}, & otherwise \end{cases} \tag{9}$$

Then, the server reserves the fused knowledge $\mathcal{P}_G^t$ in the knowledge base and distributes it to the participants of the next communication round. $\mathcal{P}_k^t$ is overwritten by $\mathcal{P}_G^t$ before the local training to realize the spatial knowledge transfer.

6. Experiments

In this section, we consider a realistic FCIL training scenario composed of an alternate local class-incremental update and global federated iteration. We conduct comprehensive experiments under synchronous and asynchronous FCIL settings to evaluate FedProK in terms of accuracy and trustworthiness.

6.1. Experiment Setting

Datasets. We conduct experiments on two datasets: (1) CIFAR-10 [19]: 10 classes, 32×32 pixels images, 5000 training and 1000 test images per class. (2) CIFAR-100 [19]: 100 classes, 32×32 pixels images, 500 training and 100 test images per class.

Synchronous FCIL Setting. In the synchronous setting, clients share the same class-incremental task sequence but suffer a distribution-based label imbalance, which is controlled with the Dirichlet concentration parameter α. We partitioned the whole dataset into T tasks and apportioned them to N clients according to α. For CIFAR-10, we set $N = 3, T = 2$ and $N = 3, T = 5$, each task contains data of 5 or 2 classes and $\alpha = \{0.5, 1\}$. For CIFAR-100, we set $N = 3, T = 5$, $N = 3, T = 10$, each task contains data of 20 or 10 classes and $\alpha = \{0.5, 1\}$. Specifically, not every communication round involves the arrival of new tasks. We set the interval of incremental states to be every 5 round for CIFAR-10 and every 10 round for CIFAR-100.

Asynchronous FCIL Setting. In the asynchronous setting, some of the classes are accessible to all clients, while others are private to a single client. We refer to the rate of common classes as **consensus rate**, denoted with γ. Intuitively, the smaller γ indicates a higher degree of heterogeneity. Due to the limited number of classes in CIFAR-10, we only conduct asynchronous FCIL experiments on CIFAR-100. We set $N = 3, T = 5, \gamma = 0.2$ and $N = 3, T = 5, \gamma = 0.5$, which means 10 or 25 classes are common on all clients while the rest of 40 or 25 classes are unique on one client. We set the interval of incremental states to be every 10 round under this setting.

Baselines. We compare FedProK with static FL baselines, existing FCIL methods, and hybrid methods, i.e., combination of FL and CL methods. Static FL methods include FedAvg [27], FedProx [20] and FedNova [34]. FCIL competitors include GLFC [8], FedSpace [31] and TARGET [43]. Hybrid competitors include FedEWC [18] and FediCaRL [30]. More detailed descriptions are in the Appendix.

Evaluation Metric. We utilize the final accuracy of the global model, denoted as $\mathcal{A}_G^R$, as the metric to showcase the performance of our model and baselines. This choice aligns with the primary goal of FCIL, which focuses on training a global model capable of distinguishing between all classes. The trustworthiness is assessed with $U(\theta_G)$, $P(\theta_G)$ and $E(\theta_G)$. Definitions of the metrics are in Sec. 4.

Implementation Details. The code of our method and baselines is implemented with PyTorch. The backbone network is ResNet-18. We conducted experiments on each dataset using three random seeds (42, 1999, 2024) and averaged the results. The size of the exemplar set for all of the exemplar-based methods is 500. All the experiments were conducted sequentially on NVIDIA RTX-3090 GPU.

6.2. Experimental Results

6.2.1 Results on Accuracy

Fig. 3 and Fig. 4 illustrate the test accuracy of the global model on all learned tasks after training on the current task. As shown in Fig. 3 (left), although CIFAR-10 has a limited number of total classes, the global models still suffer significant performance decline with the arrival of new classes. Despite the initial round, FedProK consistently outperforms other FCIL methods by the margin of $9.22\% \sim 44.08\%$ when $\alpha = 1$ and $5.42\% \sim 47.57\%$ when $\alpha = 0.5$. In Fig. 3 (right), FedEWC completely failed in preventing catastrophic forgetting of the previous tasks, while FedSpace and FediCaRL are just slightly better. Our method outperforms the competitive GLFC and TARGET by $4.14\% \sim 7.72\%$ when $\alpha = 1$ and $4.72\% \sim 8.18\%$ when $\alpha = 0.5$. Fig. 4 illustrates the results on CIFAR-100 with 10 incremental tasks when $\alpha = 1$ and $\alpha = 0.5$, in which our FedProK maintains the highest test accuracy during the entire training.

The final accuracy of the global model on CIFAR-10 and CIFAR-100 are reported in Tab. 1. The level of data heterogeneity among clients gradually increases from the left to the right end of the table, and the performance of each method correspondingly decreases. Static FL methods (FedAvg, FedProx, and FedNova) failed in both synchronous FCIL and asynchronous FCIL settings, indicating that the dynamically incremented classes adversely impact the model performance. The regularization-based hybrid

Setting	Synchronous FCIL								Asynchronous FCIL	
Dataset	CIFAR-10				CIFAR-100					
Method	T=2		T=5		T=5		T=10		T=5	
	$\alpha = 1$	$\alpha = 0.5$	$\alpha = 1$	$\alpha = 0.5$	$\alpha = 1$	$\alpha = 0.5$	$\alpha = 1$	$\alpha = 0.5$	$\gamma = 0.5$	$\gamma = 0.2$
FedAvg	43.59	41.17	18.45	16.53	16.42	15.92	8.66	8.40	2.78	1.03
FedEWC	43.13	40.05	17.92	15.22	16.86	16.06	8.84	8.04	3.57	1.02
FediCaRL	60.01	58.14	26.81	25.17	24.25	20.51	18.50	14.14	9.66	2.28
FedProx	46.27	43.55	17.49	17.05	19.43	19.59	10.82	9.12	3.23	1.97
FedNova	46.20	44.17	19.61	18.14	18.97	19.88	11.01	8.96	4.13	1.15
GLFC	72.34	67.99	55.80	52.74	36.88	31.43	26.19	24.54	13.59	9.92
FedSpace	55.15	50.60	21.60	19.56	21.23	21.35	19.00	18.24	4.15	3.76
TARGET	76.52	74.91	59.66	57.37	34.89	33.33	22.85	20.71	14.17	11.67
Ours-w/oFT	43.74	41.10	17.73	16.25	16.55	15.73	8.76	8.28	-	-
Ours-w/oPKF	78.34	76.73	58.76	56.27	35.23	34.88	24.90	24.17	-	-
Ours (FedProK)	**87.62**	**84.13**	**64.34**	**62.79**	**41.02**	**39.61**	**26.94**	**26.07**	**18.52**	**12.21**

Table 1. Final accuracy (%) of the global model on CIFAR-10 and CIFAR-100, where α denotes the Dirichlet concentration parameter in synchronous setting and γ denotes the consensus rate in asynchronous setting. The degree of heterogeneity increases from the left to the right end of the table.

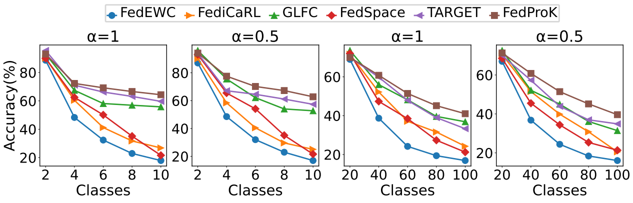


Figure 3. Experimental results on CIFAR-10 (left) and CIFAR-100 (right) with 5 incremental tasks, where α denotes the Dirichlet concentration parameter.

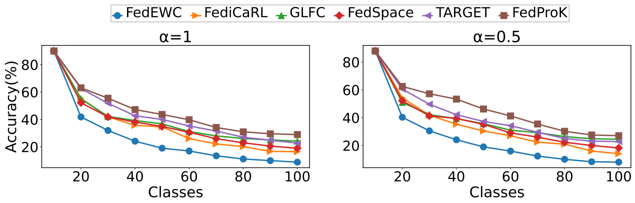


Figure 4. Experimental results on CIFAR-100 with 10 incremental tasks, where α denotes the Dirichlet concentration parameter.

method (FedEWC) showed even poorer performance than heterogeneous FL methods (FedProx and FedNova), implying that EWC is incapable of FCIL. We observed that our model outperformed existing FCIL methods (GLFC, FedSpace, and TARGET) under all settings. FedProK also achieved better accuracy by $8.54\% \sim 0.54\%$ under the most challenging scenario, i.e., asynchronous FCIL with $\gamma = 0.2$. The experimental results highlight the effectiveness of our method in overcoming catastrophic forgetting and data heterogeneity.

6.2.2 Results on Trustworthiness

We compared the trustworthiness of our method with baselines in terms of continual utility, efficiency, and privacy. As stated in Sec. 4, we validate the continual utility of our method and baselines with the trade-off between stability and plasticity, which are measured with the global accuracy of the previous classes and current classes respectively. Efficiency is measured with the average training time per communication round, reflecting the total cost of communication and computation of federated training. Privacy is measured by the similarity between the reconstructed images generated by malicious attackers and the original images, where a larger result indicates better performance in privacy preservation.

Based on the experimental results reported in Tab. 2, FedProK exhibited superior trustworthiness compared with various baselines. Specifically, our method outperformed baselines on continual utility by $7.46\% \sim 31.54\%$ when $\lambda = 0.5$. And FedProK showed better efficiency than GLFC and TARGET, with 47.29 and 24.06 seconds per communication round, respectively. Besides, FedProK also outperformed in data privacy by $25.0549 \sim 13.9017$.

Fig. 5 illustrates the reconstructed images of malicious attackers [50]. As an exemplar-based method, FediCaRL suffers the most significant data leakage where the private

Method	Continual Utility(%)	Efficiency (s)	Privacy (10^{-3})
FedEWC	21.69	**79.72**	2.4763
FediCaRL	31.37	94.27	0.5569
GLFC	45.04	165.33	11.7101
TARGET	45.77	142.10	-
Ours (FedProK)	**53.23**	118.04	**25.6118**

Table 2. Continual utility, efficiency, and privacy of our method and baselines on CIFAR-100 with 5 class-incremental states, which are measured with Eq. (1) ($\lambda = 0.5$), training time per epoch, and whether the FCIL method is exemplar-based.

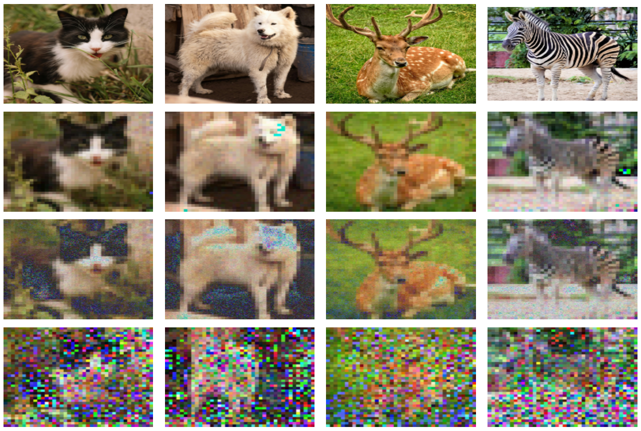
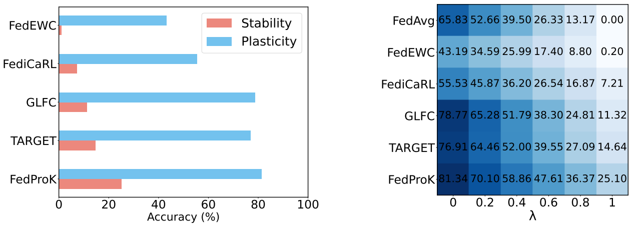


Figure 5. Illustration of the reconstructed images of malicious attackers, where the first line is the ground truth, the second line is the reconstructed images of FediCaRL, the third line is the reconstructed images of GLFC and the bottom line is the reconstructed images of our method.

images are completely reconstructed by attackers. GLFC exhibits some robustness against attacks because it adds Gaussian noise to the stored exemplars [8]. The reconstructed images of FedProK show the greatest differences with the ground-truth images, affirming the superiority of our exemplar-free approach to privacy preservation.

Fig. 6a illustrates the stability and plasticity results separately, indicating that our method achieved state-of-the-art performance on both. Hybrid methods (FedEWC and FediCaRL) completely failed in achieving excellent results on continual utility because of the unacceptably poor stability. FCIL methods (GLFC and TARGET) showed generally better continual utility than hybrid methods.

Additionally, Continual utility results under different values of λ shown in Fig. 6b demonstrate that FedProK struck a more favorable balance between stability and plasticity under different λ, thereby ensuring equilibrium in discriminative capabilities for both previous and new classes.



(a) Stability and plasticity.

(b) Continual utility.

Figure 6. Continual utility of FedProK and baselines on CIFAR-100 with 5 incremental tasks. (a) illustrates the stability and plasticity separately of our model and baselines. (b) illustrates the continual utility as a measurement of the stability-plasticity trade-off of our model and baselines, where λ is the hyperparameter. A larger λ indicates more emphasis on stability.

6.3. Ablation Study

As shown in Tab. 1, the effects of each module in our method are validated via ablation studies. Ours-w/oFT and Ours-w/oPKF denote the performance of FedProK without feature translation (FT) on the clients and prototypical knowledge fusion (PKF) on the server. Note that if FT is removed, PKF will not have any effect. Compared with the entire version of FedProK, Ours-w/oFT and Ours-w/oPKF suffer significant performance degradation under all settings. The performance degradation of **Ours-w/o PKF** supports that the PKF module successfully mitigates data heterogeneity and achieves open-world continual learning by exchanging prototypical knowledge selectively on the server. The performance degradation of **Ours-w/oFT** supports the overall effectiveness of FedProK in spatial-temporal knowledge transfer by geometrically translating the feature distribution of current classes to that of previous classes and replaying the translated feature sets in subsequent local training.

7. Conclusion

In this work, we formulate a trustworthy FCIL framework with regard to continual utility, efficiency and privacy objectives. We propose FedProK as a solution to the challenges posed by trustworthy FCIL. Specifically, FedProK consists of two components: (1) a client-side feature translation procedure that generates pseudo representations of previous classes to achieve temporal knowledge transfer, and (2) a server-side prototypical knowledge fusion mechanism that shares global prototype lists among participants to achieve spatial knowledge transfer. We conduct experiments in both synchronous and asynchronous settings to demonstrate that our FedProK outperforms the other state-of-the-art methods in terms of accuracy and trustworthiness on common benchmark datasets, which validates the effectiveness of our FedProK in knowledge transfer.

References

[1] Sawsan AbdulRahman, Hanine Tout, Hakima Ould-Slimane, Azzam Mourad, Chamseddine Talhi, and Mohsen Guizani. A survey on federated learning: The journey from centralized to distributed on-site learning and beyond. *IEEE Internet of Things Journal*, 8(7):5476–5497, 2020. 1

[2] Manoj Ghuhan Arivazhagan, Vinay Aggarwal, Aaditya Kumar Singh, and Sunav Choudhary. Federated learning with personalization layers. *arXiv preprint arXiv:1912.00818*, 2019. 3

[3] Gaurav Bagwe, Xiaoyong Yuan, Miao Pan, and Lan Zhang. Fed-cprompt: Contrastive prompt for rehearsal-free federated continual learning. In *Federated Learning and Analytics in Practice: Algorithms, Systems, Applications, and Opportunities*, 2023. 2

[4] Di Chai, Leye Wang, Liu Yang, Junxue Zhang, Kai Chen, and Qiang Yang. A survey for federated learning evaluations: Goals and measures. *arXiv preprint arXiv:2308.11841*, 2023. 1, 4

[5] Marcos F Criado, Fernando E Casado, Roberto Iglesias, Carlos V Regueiro, and Senén Barro. Non-iid data and continual learning processes in federated learning: A long road ahead. *Information Fusion*, 88:263–280, 2022. 2

[6] Shenghong Dai, Bryce Yicong Chen, Jy-yong Sohn, SM Iftekharul Alam, Ravikumar Balakrishnan, Suman Banerjee, Nageen Himayat, and Kangwook Lee. Fedgp: Buffer-based gradient projection for continual federated learning. In *Federated Learning Systems (FLSys) Workshop@ MLSys 2023*, 2023. 1

[7] Matthias De Lange, Rahaf Aljundi, Marc Masana, Sarah Parisot, Xu Jia, Aleš Leonardis, Gregory Slabaugh, and Tinne Tuytelaars. A continual learning survey: Defying forgetting in classification tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(7):3366–3385, 2021. 3, 6

[8] Jiahua Dong, Lixu Wang, Zhen Fang, Gan Sun, Shichao Xu, Xiao Wang, and Qi Zhu. Federated class-incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10164–10173, 2022. 1, 2, 3, 4, 6, 8

[9] Yi Feng, Chuanyi Li, and Vincent Ng. Legal judgment prediction: A survey of the state of the art. *IJCAI. ijcai. org*, pages 5461–9, 2022. 1

[10] Hanlin Gu, Lixin Fan, Yan Kang Bowen Li, Yuan Yao, and Qiang Yang. A novel approach to simultaneously improve privacy, efficiency and reliability of federated dnn learning. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence*, 2022. 3

[11] Yuting He, Yiqiang Chen, Xiaodong Yang, Hanchao Yu, Yi-Hua Huang, and Yang Gu. Learning critically: Selective self-distillation in federated learning on non-iid data. *IEEE Transactions on Big Data*, 2022. 2

[12] Zeou Hu, Kiarash Shaloudegi, Guojun Zhang, and Yaoliang Yu. Federated learning meets multi-objective optimization. *IEEE Transactions on Network Science and Engineering*, 9 (4):2039–2051, 2022. 3

[13] Yaochu Jin, Hangyu Zhu, Jinjin Xu, and Yang Chen. Evolutionary multi-objective federated learning. In *Federated Learning: Fundamentals and Advances*, pages 139–164. Springer, 2022. 2

[14] Jiawen Kang, Zehui Xiong, Dusit Niyato, Yuze Zou, Yang Zhang, and Mohsen Guizani. Reliable federated learning for mobile networks. *IEEE Wireless Communications*, 27(2): 72–80, 2020. 2

[15] Yan Kang, Jiahuan Luo, Yuanqin He, Xiaojin Zhang, Lixin Fan, and Qiang Yang. A framework for evaluating privacy-utility trade-off in vertical federated learning. *arXiv preprint arXiv:2209.03885*, 2022. 3

[16] Yan Kang, Hanlin Gu, Xingxing Tang, Yuanqin He, Yuzhu Zhang, Jinnan He, Yuxing Han, Lixin Fan, and Qiang Yang. Optimizing privacy, utility and efficiency in constrained multi-objective federated learning. *arXiv preprint arXiv:2305.00312*, 2023. 3

[17] Dongwan Kim and Bohyung Han. On the stability-plasticity dilemma of class-incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20196–20204, 2023. 2

[18] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114(13):3521–3526, 2017. 2, 6

[19] Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. The cifar-10 dataset. *online: http://www. cs. toronto. edu/kriz/cifar. html*, 55(5), 2014. 6

[20] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. *Proceedings of Machine Learning and Systems*, 2:429–450, 2020. 3, 6

[21] Tao Lin, Lingjing Kong, Sebastian U Stich, and Martin Jaggi. Ensemble distillation for robust model fusion in federated learning. *Advances in Neural Information Processing Systems*, 33:2351–2363, 2020. 3

[22] Yang Liu, Yan Kang, Tianyuan Zou, Yanhong Pu, Yuanqin He, Xiaozhou Ye, Ye Ouyang, Ya-Qin Zhang, and Qiang Yang. Vertical federated learning: Concepts, advances and challenges. *arXiv preprint arXiv:2211.12814*, 2022. 1

[23] Sin Kit Lo, Yue Liu, Qinghua Lu, Chen Wang, Xiwei Xu, Hye-Young Paik, and Liming Zhu. Toward trustworthy ai: Blockchain-based architecture design for accountability and fairness of federated learning systems. *IEEE Internet of Things Journal*, 10(4):3276–3284, 2022. 2

[24] Kangyang Luo, Xiang Li, Yunshi Lan, and Ming Gao. Gradma: A gradient-memory-based accelerated federated learning with alleviated catastrophic forgetting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3708–3717, 2023. 2

[25] Lingjuan Lyu, Han Yu, and Qiang Yang. Threats to federated learning: A survey. *arXiv preprint arXiv:2003.02133*, 2020. 1

[26] Yuhang Ma, Zhongle Xie, Jue Wang, Ke Chen, and Lidan Shou. Continual federated learning based on knowledge dis-

tillation. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence*, 2022. 2

[27] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial Intelligence and Statistics*, pages 1273–1282. PMLR, 2017. 1, 6

[28] Viraaji Mothukuri, Reza M Parizi, Seyedamin Pouriyeh, Yan Huang, Ali Dehghantanha, and Gautam Srivastava. A survey on security and privacy of federated learning. *Future Generation Computer Systems*, 115:619–640, 2021. 1

[29] Grégoire Petit, Adrian Popescu, Hugo Schindler, David Picard, and Bertrand Delezoide. Fetril: Feature translation for exemplar-free class-incremental learning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3911–3920, 2023. 2, 5

[30] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. icarl: Incremental classifier and representation learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2001–2010, 2017. 2, 6

[31] Donald Shenaj, Marco Toldo, Alberto Rigon, and Pietro Zanuttigh. Asynchronous federated continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5054–5062, 2023. 6

[32] Neta Shoham, Tomer Avidor, Aviv Keren, Nadav Israel, Daniel Benditkis, Liron Mor-Yosef, and Itai Zeitak. Overcoming forgetting in federated learning on non-iid data. *arXiv preprint arXiv:1910.07796*, 2019. 2

[33] Izzeddin Teeti, Salman Khan, Ajmal Shahbaz, Andrew Bradley, and Fabio Cuzzolin. Vision-based intention and trajectory prediction in autonomous vehicles: A survey. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22, Lud De Raedt, Ed*, pages 5630–5637, 2022. 1

[34] Jianyu Wang, Qinghua Liu, Hao Liang, Gauri Joshi, and H Vincent Poor. Tackling the objective inconsistency problem in heterogeneous federated optimization. *Advances in Neural Information Processing Systems*, 33:7611–7623, 2020. 6

[35] Yunhui Wang, Weichu Zheng, Zifei Liu, Jinyan Wang, Hongjian Shi, Mingyu Gu, and Yicheng Di. A federated network intrusion detection system with multi-branch network and vertical blocking aggregation. *Electronics*, 12(19):4049, 2023. 3, 4

[36] Guoyizhe Wei and Xiu Li. Knowledge lock: Overcoming catastrophic forgetting in federated learning. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 601–612. Springer, 2022. 2

[37] Guile Wu, Shaogang Gong, and Pan Li. Striking a balance between stability and plasticity for class-incremental learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1124–1133, 2021. 2

[38] Qiang Yang, Yang Liu, Tianjian Chen, and Yongxin Tong. Federated machine learning: Concept and applications. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 10(2):1–19, 2019. 4

[39] Xin Yang, Hao Yu, Xin Gao, Hao Wang, Junbo Zhang, and Tianrui Li. Federated continual learning via knowledge fusion: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 2024. 1, 2, 3

[40] Zhanpeng Yang, Yuanming Shi, Yong Zhou, Zixin Wang, and Kai Yang. Trustworthy federated learning via blockchain. *IEEE Internet of Things Journal*, 10(1):92–109, 2022. 2

[41] Jaehong Yoon, Wonyong Jeong, Giwoong Lee, Eunho Yang, and Sung Ju Hwang. Federated continual learning with weighted inter-client transfer. In *International Conference on Machine Learning*, pages 12073–12086. PMLR, 2021. 1, 2, 3

[42] Chen Zhang, Yu Xie, Hang Bai, Bin Yu, Weihong Li, and Yuan Gao. A survey on federated learning. *Knowledge-Based Systems*, 216:106775, 2021. 1

[43] Jie Zhang, Chen Chen, Weiming Zhuang, and Lingjuan Lyu. Target: Federated class-continual learning via exemplar-free distillation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4782–4793, 2023. 1, 2, 6

[44] Xiaojin Zhang, Hanlin Gu, Lixin Fan, Kai Chen, and Qiang Yang. No free lunch theorem for security and utility in federated learning. *ACM Transactions on Intelligent Systems and Technology*, 14(1):1–35, 2022. 3

[45] Xiaojin Zhang, Yan Kang, Kai Chen, Lixin Fan, and Qiang Yang. Trading off privacy, utility and efficiency in federated learning. *ACM Transactions on Intelligent Systems and Technology*, 2022. 3

[46] Xiaojin Zhang, Yan Kang, Kai Chen, Lixin Fan, and Qiang Yang. Trading off privacy, utility, and efficiency in federated learning. *ACM Trans. Intell. Syst. Technol.*, 2023. 3

[47] Bowen Zhao, Xi Xiao, Guojun Gan, Bin Zhang, and Shu-Tao Xia. Maintaining discrimination and fairness in class incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13208–13217, 2020. 2

[48] Xingying Zhao and Dong Shen. Fedsw: Federated learning with adaptive sample weights. *Information Sciences*, 654: 119873, 2024. 2

[49] Hangyu Zhu and Yaochu Jin. Multi-objective evolutionary federated learning. *IEEE Transactions on Neural Networks and Learning Systems*, 31(4):1310–1322, 2019. 2

[50] Ligeng Zhu, Zhijian Liu, and Song Han. Deep leakage from gradients. *Advances in neural information processing systems*, 32, 2019. 7