ProtoHAR: Prototype Guided Personalized Federated Learning for Human Activity Recognition

Dongzhou Cheng, Lei Zhang, Can Bu, Xing Wang, Hao Wu, and Aiguo Song, *Senior Member, IEEE*

***Abstract*—Federated Learning (FL) has recently attracted great interest in sensor-based human activity recognition (HAR) tasks. However, in real-world environment, sensor data on devices is non-independently and identically distributed (Non-IID), e.g., activity data recorded by most devices is sparse, and sensor data distribution for each client may be inconsistent. As a result, the traditional FL methods in the heterogeneous environment may incur a drifted global model that causes slow convergence and a heavy communication burden. Although some FL methods are gradually being applied to HAR, they are designed for overly ideal scenarios and do not address such Non-IID problem in the real-world setting. It is still a question whether they can be applied to cross-device FL. To tackle this challenge, we propose ProtoHAR, a prototype-guided FL framework for HAR, which aims to decouple the representation and classifier in the heterogeneous FL setting efficiently. It leverages the global prototype to correct the activity feature representation to make the prototype knowledge flow among clients without leaking privacy while solving a better classifier to avoid excessive drift of the local model in personalized training. Extensive experiments are conducted on four publicly available datasets: USC-HAD, UNIMIB-SHAR, PAMAP2, and HARBOX, which are collected in both controlled environments and real-world scenarios. The results show that compared with the state-of-the-art FL algorithms, ProtoHAR achieves the best performance and faster convergence speed in HAR datasets.**

***Index Terms*—Human activity recognition, federated learning, representation learning, sensor, deep learning.**

Manuscript received 20 November 2022; revised 4 April 2023; accepted 8 May 2023. Date of publication 11 May 2023; date of current version 7 August 2023. The work was supported in part by the National Nature Science Foundation of China under Grant 61962061 and in part by the Natural Science Foundation of Jiangsu Province under Grant BK20191371. *(Corresponding authors: Lei Zhang; Hao Wu.)*

Dongzhou Cheng, Lei Zhang, Can Bu, and Xing Wang are with the School of Electrical and Automation Engineering, Nanjing Normal University, Nanjing 210023, China (e-mail: chengdongzhou666@qq.com; leizhang@njnu.edu.cn; bucanya123@gmail.com; wangxing@qq.com).

Hao Wu is with the School of Information Science and Engineering, Yunnan University, Kunming 650500, China (e-mail: haowu@ynu.edu.cn).

Aiguo Song is with the School of Instrument Science and Engineering, Southeast University, Nanjing 210096, China (e-mail: a.g.song@seu.edu.cn).

Digital Object Identifier 10.1109/JBHI.2023.3275438

I. Introduction

TODAY, the popularity of cell phones and smart bracelets has made them easier to collect sensor data, which has greatly stimulated researches for sensor-based Human Activity Recognition (HAR) [1]. Due to its ubiquitous applications in security, monitoring, health management, etc. [2], privacy concerns are growing. As more countries are updating their data protection laws, the privacy protection issue has recently received great interest in HAR [2]. Generally speaking, the higher one sensor's inference potential, the less eager a person is to release personal private data. Since each client performs activities in a unique behavior style, adversaries might infer user-sensitive information such as age and gender from the time-series sensor data. Specifically, for deep learning models, its black-box nature may inadvertently reveal user-discriminative features. Due to privacy protection, data is typically distributed among different devices, and the clients would be not willing to send their own private data to a central server for training a model, which inevitably restricts the activity recognition ability of smart wearable devices.

This fundamental obstacle has been well addressed by Federated Learning (FL) [3], which is widely employed to tackle above HAR challenges [4]. The FedAvg algorithm is first introduced in [5], which leverages multiple clients to minimize the local empirical risk via collaboratively learning a shared global model without relying on raw data. Though FL has achieved remarkable success in many areas, it still remains an important challenge for heterogeneous HAR problem [4]. For example, in the case of personalized cross-device FL [6] for HAR, the sensor data from different clients stored in their personal mobile devices usually requires electronic health records to be mined. First, the sensor data recorded by different clients is usually inconsistent, due to their behavior habits or physical characteristics (e.g., age, gender, height). Second, there often exists label skew across different clients [7]. For example, a cyclist might have many *cycling* samples but few *walking* samples. This results in data heterogeneity [8], [9], which makes the local optimal points of clients be always inconsistent with the global optimal point [10] (i.e., *Client-Drift*). Due to data inconsistency, FedAvg does not guarantee convergence and only leads to suboptimal performance. It has been proved theoretically [11] and experimentally [9] that traditional FL does not work well in such heterogeneous scenario. Thus, learning a global model may not be ideal for the *Client-Drift* problem.

2168-2194 © 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: NANJING NORMAL UNIVERSITY. Downloaded on August 08,2023 at 01:24:53 UTC from IEEE Xplore. Restrictions apply.

Various studies have been conducted to address the problem of data heterogeneity in HAR, primarily from two complementary perspectives: one is to improve the effectiveness of model aggregation, where cluster FL is a popular method [12], [13]. In cluster FL, the cloud server aggregates a handful of specialized models for similar clients, which allows a trade-off between personalization and generalization to improve activity recognition performance. However, there is a too strong assumption because such methods usually require the participation of all clients so as to obtain similarity information among them. Considering the fact that mobile devices are often offline, it could not be applied to wearable HAR tasks in the real-world scenario. The other is to focus on stabilizing local training for personalization by fine-tuning a single global model [14], [15]. However, this strategy fails to take advantage of the underlying knowledge among clients, whose diversity would potentially reveal interior structure of local data and hence merits further research.

Inspired by the concept of prototype learning [16], we conjecture that averaging feature representations from different data distributions on heterogeneous clients can form latent prototype knowledge. For example, when different clients perform the same *walking* activity, they usually have slightly distinct sensor recordings of *walking* due to different behavior patterns caused by gender, age, height, etc. In other words, their prototypes from different clients would be slightly diverse for the same activity category. Despite this, it is argued that these sensor recordings might share a common *walking* prototype, which is due to that the vast majority of those clients still share somewhat similar characteristics for the same activity class. Intuitively, exchanging those concept-specific prototypes among different clients could enable them to obtain more knowledge about the concept of *walking*. Motivated by this, in this article, we present a novel prototype aggregation-based HAR approach in heterogeneous FL scenario. Our main idea is to exchange prototypes and a global representation network among clients and a central server, which does not require model parameters or gradients to be aggregated. In this case, different clients may exchange information via sharing their prototypes, each of which can be seen as one class by the mean value transformed from the observed samples belonging to the same activity category. Since the parameters of the local prototypes are not larger than that of the classifier, ProtoHAR does not incur additional communication overhead compared to the current FL framework. Specifically, ProtoHAR learns a set of generalized global prototypes derived solely from clients' local activity prototypes. These prototypes, which contain pooled knowledge from other peer clients, are later broadcasted to all clients, escorting their own local representation training over the latent space. To implement a personalized model, our key idea is to *minimize the empirical risk of the classifier on the representation guided by the global prototype*. That is, a more robust representation naturally leads to a better optimization direction for the classifier, which can not only capture the common knowledge among clients but also handle data heterogeneity. Overall, our main contributions can be summarized as follows:

First, in this article, borrowing the concept prototype learning, we propose a new prototype-based FL framework to tackle challenging HAR problem with statistical heterogeneity, where both prototypes and representations are transmitted between the server and clients. To our knowledge, this paper is the first prototype aggregation-based algorithm for activity recognition in the heterogeneous FL scenario. In particular, prototypes are utilized to refine the global representation so as to decouple feature representation and classifier effectively.

Second, different from prior most works focusing on gradient-based aggregation, the proposed algorithm requires no model parameters or gradients to be aggregated. Instead, it exchanges information via sharing prototypes and representations, where each abstract prototype can be seen as an activity class by the mean representations transformed from the observed samples belonging to the same activity category. Compared to gradient-based aggregation, aggregating the prototypes and representations can facilitate more efficient communication among heterogeneous clients.

Third, we perform extensive experiments by comparing our proposed ProtoHAR with eight current state-of-the-art FL algorithms on four public HAR benchmarks, which show that it can significantly outperform other competitive FL baselines while accelerating convergence and reducing communication burden to a certain extent.

II. Related Work

A. FL in Non-IID Scenario

In fact, FL has been a mainstream strategy to address current challenges in privacy concerns [5], which could collaboratively learn a shared model without requiring raw data collected from users. During recent years, there has been a lot of recent researches that focus on the communication challenges in FL scenario with statistical heterogeneity, i.e., how to reduce the communication cost. For example, Paragliola et al. [17] define a novel FL strategy aimed at reducing the communication costs through the transmission of a subpart of the local model instead of the whole model. A comprehensive analysis is provided to evaluate the trade-off between the communication cost and performance. Zhao et al. [18] propose a data-sharing strategy to improve FedAvg with Non-IID data via distributing a small amount of globally shared data containing examples from each class, which leads to a trade-off between centralization and accuracy. Duan et al. [19] introduce a self-balancing FL framework named Astraea to handle the class imbalance issue through run-time data augmentation. However, these previous works have not considered the issue caused by an increasing size in continuous data over time, and little effort has been devoted to such non-stationary data stream. To address the issue of continuous learning, Paragliola et al. [20] for the first time extend a variant of previous FedAvg algorithm in a non-stationary scenario, and then investigate the extent to which the catastrophic forgetting problem cause by dynamic clients in the FL setting. In fact, such statistical heterogeneity among multiple client nodes always remains an important challenge for HAR in the FL scenario. Most prior works [5], [8], [10], [21] primarily focus on the sole heterogeneous scenario, all of which are based on an idea of gradient aggregation, hence raising concerns about communication efficiency and gradient-based attacks. In general, the concept of prototypes [16] (i.e., the mean of multiple features) has been widely used to resolve

Authorized licensed use limited to: NANJING NORMAL UNIVERSITY. Downloaded on August 08,2023 at 01:24:53 UTC from IEEE Xplore. Restrictions apply.

image classification problem with a limited number of training examples. Intuitively, the learning scenario is in well line with the latent assumption of a cross-client FL setting, where each client only has a limited number of activity samples to train a personalized model independently for the desired HAR performance. However, prototypes have been rarely explored in a large variety of HAR tasks. Different from prior most works focusing on gradient-based aggregation, we primarily seek to combine prototype aggregation and representation learning for HAR in such heterogeneous FL setting.

B. FL on Human Activity Recognition

During the past decade, several mainstream deep neural networks such as convolutional networks, LSTM, residual networks, and autoencoders [2], [22], [23] have been widely applied to address sensor-based HAR problem. However, prior most works primarily focus on training a single global model by aggregating raw data from all clients in a central server, which lack privacy protection and personalization for each client [2]. FL has been employed in wearable HAR to distribute the training process of the model to all participating devices [12], [13], [14], [15], [24]. Xiao et al. [25] design a perceptive extraction network (PEN) for HAR, which combines the federated averaging algorithm to improve the performance of PEN. By building two models, i.e., a deep neural network and a softmax regression, Sozinov et al. [4] apply FL to train two HAR classifiers and compare their performance to centralized learning. However, the aforementioned works are impractical in real-world scenario since they neglect that the client devices might adapt to the user behavior and habit due to personalization, which is a crucial part of HAR [26]. To address this challenge, Chen et al. [14] adopt a transfer learning method on client side to improve personalization for Parkinson's disease auxiliary diagnosis. Ouyang et al. [12] propose a multi-task federated clustering method called ClusterFL for HAR, which allows similar nodes for collaborative learning. Wu et al. [15] propose FedHome by designing a generative convolutional autoencoder to reconstruct a class-balanced dataset, which is used to fine-tune the activity recognition model on edge device. However, this method still requires a large number of communication rounds to obtain a stable global model, and the clients cannot be personalized at an early stage during the FL process. Those approaches are still based on a single global model without exploring how to decouple representation and classifier to address the Non-IID problem more efficiently and effectively, meanwhile bringing additional memory consumption, which cannot be directly applied to tiny wearable devices in the scenario of IoT (Internet of Things). In addition, prior most works assume that all clients participate in training during each communication round, which does not consider overall communication efficiency in real-world scenarios. Compared to existing FL methods in HAR, ProtoHAR first proposes an efficient prototype-based solution without excessive communication burden and memory footprint, which improves the representation quality of activity features, thus achieving higher accuracy for personalized classifiers in HAR tasks.

C. Prototype Learning

Prototypical Network(PN) [16], [27] takes the mean feature as the class prototype and leverages Euclidean distance to restrict data points to their nearest prototype in an embedded space. Prototypes have been widely employed in various image classification tasks [28], [29]. In this article, we utilize the concept of prototypes to describe the fundamental characteristics of activity classes, and then employ it to solve heterogeneous HAR problem. Prototype-based solution is primarily used in a scenario with the number limited samples, which is consistent with the latent assumption of cross-device FL: one client has no enough samples to train deep models. This assumption is widely supported in FL-based benchmark datasets [7], [30] and has been validated in HAR area [12], [15]. In particular, Cruciani et al. [31] have proposed a new personalization approach for HAR by combining an idea of semi-population for user adaptation. Specifically, a subset of clients is first identified as the best candidates from the available population so as to initialize one classifier for the target client, and then a semi-population neural network classifier is trained over data from the subset of clients. Finally, personalizing activity recognition could be realized by fine-tuning the classifier's weight parameters over a small amount of labeled data from the target client. Different from such a semi-population approach, our work is primarily built on an idea of prototypes. However, it can be further combined with the proposed semi-population solution by identifying a subset of prototypes as potential candidates from the available population, which has a potential to reduce the number of labeled data required for personalization while improving communication efficiency in such heterogonous FL setting.

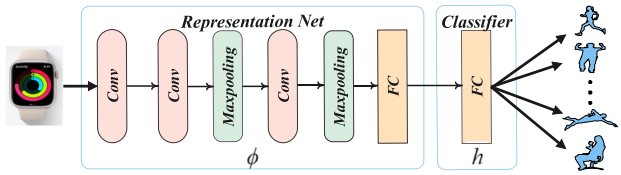


Fig. 1. Backbone classification network for HAR.

III. The ProtoHAR Framework

A. Network Architecture

As shown in Fig. 1, the backbone classification network consists of three convolutional layers, two max-pooling layers, two fully connected layers, and one softmax layer, which has been widely utilized in HAR area due to its powerful ability in extracting features automatically from raw sensor data [2], [14]. Generally speaking, such deep model can be seen as two separate parts: 1) representation layers (a.k.a. embedding functions) used to extract representation vectors from raw input. 2) decision layers are used to make a classification decision for a given representation vector. For convenience of notation, the former is denoted as **Representation Net** ϕ while the latter is called as **Classifier** h.

B. Problem Setting

Standard federated Learning: Suppose there are N clients, denoted as $C_1, \ldots, C_N$. Client C_i has a local private dataset D_i drawn from $\mathbb{P}_i(x, y)$, where x and y represent the input features and related class labels, respectively. Our goal is to learn a better global model w over the dataset $D = \bigcup_{i \in [N]} D^i$ with the help of

Authorized licensed use limited to: NANJING NORMAL UNIVERSITY. Downloaded on August 08,2023 at 01:24:53 UTC from IEEE Xplore. Restrictions apply.

a central server, where the private data are not exchanged. The objective is to solve:

$$\min_{w} \mathscr{L}(w) = \sum_{i=1}^{N} \frac{|D^i|}{|D|} L_i(w), \tag{1}$$

where $L_i(w) = \mathbb{E}_{(x,y)\sim D^i}[\ell_i(w;(x,y))]$ is the empirical loss of C_i. To avoid privacy leakage, all clients are restricted to share their raw data with each other. FedAvg [5] is proposed to coordinate multiple clients to collaboratively train a global model in a central server while preserving data privacy. In FedAvg, during each communication round, n clients are selected to optimize their local models on local data, and then the server utilizes the local model parameters $\sum_{i=1}^{n} \theta_i$ sent by the subset of clients to update the global model w:

$$w = \sum_{i=1}^{n} \frac{|D_i|}{|D|} \theta_i. \tag{2}$$

However, this would yield a solution that performs worse in heterogeneous setting where the data distribution D_i varies across clients. Indeed, several researches [7], [9], [10] argue that heterogeneous datasets lead to *Client-drift* in standard FL setting, where multiple local updates in heterogeneous setting push each client apart from the global optimum, hence undermining performance. Therefore, learning a shared model w may not provide a good solution to Problem (1).

Learning a Global Representation: From a new perceptive of FL [32], the heterogeneous data distributed across clients can share a common representation despite different labels. Through this shared (low-dimensional) representation, it is much easier to predict the labels for each client using either a linear classifier or a shallow neural network [32], [33].

Formally, we consider a setting consisting of a global representation net $\phi : \mathbb{R}^d \to \mathbb{R}^k$, which maps data points into a low-dimensional representation space k, and classifier $h_i : \mathbb{R}^k \to Y^Q$, that maps from the low-dimensional representation space k to the label space Q. We use $R_i(\phi)$ to denote the representation parameterized by ϕ, and a prediction for x can be generated by the function $G_i(h_i)$ parameterized by h_i. So the model for the i-th client can be written as $F_i(\phi, h_i) = (R_i(\phi) \circ G_i(h_i))$. Critically, $k \ll d$ means that the number of parameters each client needs to learn locally would be modest. As a result, we can assume that any client's optimal classifier for any fixed representation is easy to compute, which motivates the following re-written global objective:

$$\min_{\phi} \sum_{i=1}^{N} \frac{|D_i|}{|D|} \min_{h_i} L_i(\phi, h_i), \tag{3}$$

where $L_i(\phi, h_i) = \mathbb{E}_{(x,y)\in D_i} \ell_i(\phi, h_i; (x,y))$ is the local empirical loss on client i's local dataset D_i. Clients collaborate to learn the global representation network ϕ using all clients' data in our proposed method, while learning their personalized classifier h_i using own local data.

Prototype Definition in HAR: To restrict the deviation of the optimization direction, inspired by [10], [34], our algorithm aims to refine the representation training with the prototype of each activity class to alleviate the inconsistency between the local optimum and the global optimum.

Given that there is a set of activity prototypes $\mathbb{P} = \{P^{(1)}, P^{(2)}, \ldots\}$, we define a prototype $P^{(j)}$ to denote the j-th activity class in $\mathbb{P}$. For the i-th client, the prototype knowledge will be aggregated from the representation vectors in activity class j:

$$P_i^{(j)} = \frac{1}{|D_{i,j}|} \sum_{(x,y)\in D_{i,j}} R_i\left(\phi_i; x\right), \tag{4}$$

where x and y denote the training sample and its corresponding label respectively. That is to say, $D_{i,j}$ is comprised of training activity instances belonging to the j-th class in local dataset D_i. For instance, the local datasets D_i and D_k owned by two clients i and k might have different label distributions. Thus, above (4) is used to calculate the prototype $P_i^{(j)} \in \mathbb{R}^k$ for the j-th activity class in client i by averaging all representations corresponding to that class. It is common for an activity classification program installed in mobile clients, in which the central server has to maintain the overall activity prototypes $\mathbb{P} = \{P^{(1)}, P^{(2)}, \ldots\}$, while each client only needs to predict a few activity classes constituting a subset of $\mathbb{P}$. As shown in Fig. 2, the activity class set held by different clients would potentially vary, but allow an overlap between them.

Since the prototypes are computed by averaging feature vectors, the intra-class bias may exist between the actual computed prototypes based on sparse samples and the expected prototypes [27]. To resolve this problem, we provide an alternative version called *Reweighted Prototype:*

$$P_i^{(j)} = \sum_{k=1}^{K} \delta_{i,k}^{(j)} R_{i,k}\left(\phi_i; x_k\right),$$

$$\delta_{i,k}^{(j)} = \frac{\exp\left(\varepsilon \cdot Sim\left(R_{i,k}^{(j)}, P^{(j)}\right)\right)}{\sum_{k=1}^{K} \exp\left(\varepsilon \cdot Sim\left(R_{i,k}^{(j)}, P^{(j)}\right)\right)}, \tag{5}$$

where δ is the weight indicating the relation between the local representations and the global prototypes. ε is a scalar parameter and Sim denotes cosine similarity. The more similar $R(\cdot)$ is to P, the more weight is assigned to $R(\cdot)$. Equation (5) may be more stable than (4) in above scenario, but (4) is cheaper to compute and usually suffices practical requirement (thus all our experiments execute (4)). We will discuss (5) in ablation studies later.

C. Motivation

The scenario that different users have heterogeneous data distribution in label is omnipresent in HAR area. For example, a subject C_i who likes sports would more likely spend more time in *jogging* or *cycling* than a sedentary subject C_j. In particular, the private model would suffer from severe performance degradation on other domains with noticeably different distributions. As a result, learning a generalizable representation under *Client-drift* is technically challenging in HAR. This article is inspired by [10] that mitigates client-side drift with the help of correction vectors, as well as [21], [34] that employ contrastive learning and prototype learning respectively to close the gap between local and global representations. Intuitively, the models trained on an entire dataset restrict the distance between the local feature representations and the global prototypes, which can

Authorized licensed use limited to: NANJING NORMAL UNIVERSITY. Downloaded on August 08,2023 at 01:24:53 UTC from IEEE Xplore. Restrictions apply.

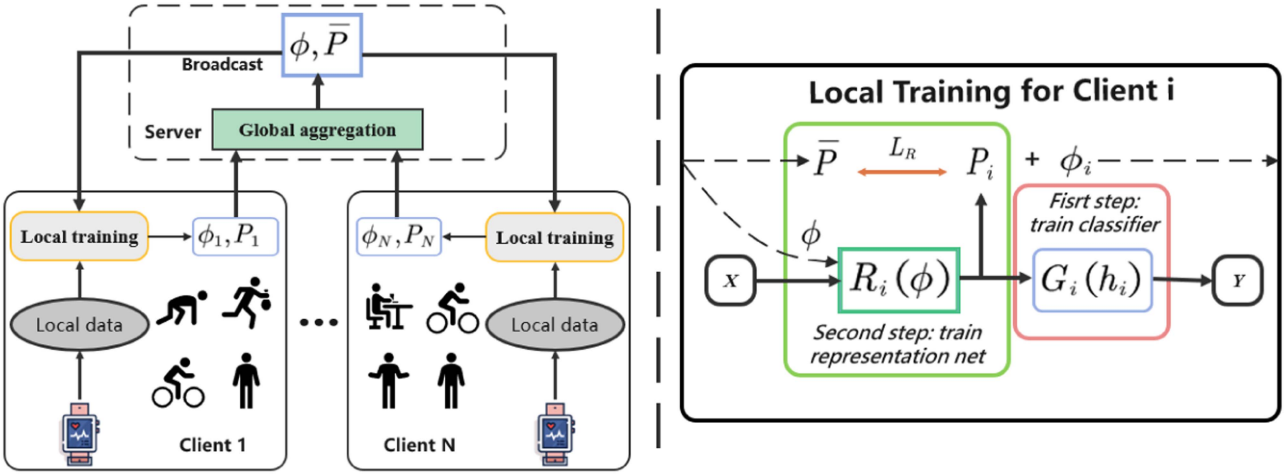


Fig. 2. Overview of the the ProtoHAR framework.

better extract activity feature representations than those trained on skewed subsets for heterogeneous HAR problem, while preserving personalized classifiers based on more generalized feature extractor so as to approximate local data distributions. Therefore, we explore how to decouple the representation and classifier for HAR more effectively in heterogeneous FL setting.

D. Proposed Method

Based on above intuition, we propose ProtoHAR, which aims to obtain a better global representation for improving the robustness of heterogeneous HAR by aggregating more generic activity prototypes. An overview of the proposed framework is shown in Fig. 2. The central server receives local representation networks $\phi_1, \phi_2, \ldots, \phi_m$ and local lightweight prototype sets $P_1, P_2 \ldots, P_m$ from m local clients, and then aggregates the global representation network and global activity prototypes, respectively. Specifically, during the local training phase, we decouple the entire training process into two steps (see the right part in Fig. 2). *First step:* the global representation is integrated into the local model to alleviate the risk of overfitting, only updating the classifier to keep the model fit for the local distribution. *Second step:* to obtain more discriminative representations, we train the representation network after freezing the classifier while restricting the distance between the local representations and global prototypes. Moreover, our algorithm maintains a lower communication cost than FedAvg [5] because the transmitted prototypes are more lightweight, whose number of parameters is $k \times q(\forall q \in [Q])$, smaller than that of classifier $(k \times Q)$ in the FedAvg model. We will detail how ProtoHAR addresses such data heterogeneity problem by combining personalization and prototype learning.

Objective in ProtoHAR: The objective of ProtoHAR is to jointly optimize a distributed learning problem and improve the personalization ability of individual classifiers by utilizing global prototypes as penalty terms in a heterogeneous HAR scenario. The server and clients attempt to learn the parameters of the global representation and the global prototype knowledge collaboratively, while the i-th client aims to learn its unique local classifier (Fig. 2). The objective of our method among heterogeneous clients can be formulated as:

$$F = \min_{\phi, \{\bar{P}^{(j)}\}_{j=1}^{|\mathbb{P}|}} \sum_{i=1}^{N} \min_{h_i} \frac{|D_i|}{|D|} L_i\left(\phi, h_i\right) + \lambda \cdot \sum_{j=1}^{|\mathbb{P}|} \sum_{i=1}^{N} \frac{|D_{i,j}|}{D_j} L_R\left(\bar{P}^{(j)}, P_i^{(j)}\right), \tag{6}$$

where L_i is the typical empirical loss (as defined in (3)), and L_R is a penalized term that measures the distance (we use L_2 distance [16]) between a local activity prototype $P_i^{(j)}$ and the corresponding global activity prototypes $\bar{P}_i^{(j)}$. λ is a hyper-parameter that controls the penalized term. Here D_j belongs to the total number of samples of the j class from the selected clients, while $D_{i,j}$ is the number of instances belonging to class j in the i-th client. The overall federated learning algorithm is shown in Algorithm 1.

Updating the global parameters: Similar to FedAvg, the server performs a weighted average of the corrected local parameters to obtain the global representation parameters:

$$\phi = \sum_{i=1}^{m} \frac{|D_i|}{|D|} \phi_i, \tag{7}$$

where m denotes the number of selected clients. Under such heterogeneous setting, some clients might have no enough training samples to obtain informative knowledge for a certain activity. ProtoHAR can aggregate prototype knowledge from different clients to produce a more generalized activity prototype. During local training, the client is trained close to the global activity prototype, which implies that when the training samples of a specific activity are insufficient for the client, the global prototype corresponding to that activity can be utilized to capture knowledge provided by other clients. Hence, the global representation network, that is aggregated from local representation networks and penalized by the global prototypes, will be more robust and can greatly alleviate the heterogeneous HAR problem.

Authorized licensed use limited to: NANJING NORMAL UNIVERSITY. Downloaded on August 08,2023 at 01:24:53 UTC from IEEE Xplore. Restrictions apply.

Algorithm 1: The ProtoHAR Framework.

Input: number of communication rounds T, number of clients N, number of selected client m, number of local updates E_h and E_ϕ, regularization coefficient λ, learning rate α and β, scale parameter τ

Server executes:
Initialize $\phi^0, h_1^0, \ldots, h_n^0$
for $t = 0, 1, \ldots, T-1$ **do**
  Sample the active client set $C_m \subseteq [N]$
  for *each client* $i \in C_m$ *in parallel* **do**
    Send the representation net ϕ^t and the global prototypes $\bar{P}^t$ to C_i
    $\phi_i^t, P_i^t \leftarrow$ **LocalTraining**$(\phi_i^t, \bar{P}^t)$
  $\phi^{t+1} \leftarrow \sum_{i=1}^{m} \frac{|D^i|}{|D|} \phi_i^t$
  $\bar{P}^{t+1} \leftarrow$ Update global prototype by Eq. (7)

Local Training:
$\phi_i^{t-1} \leftarrow \phi^t, \bar{P}^{t-1} \leftarrow \bar{P}^t$
for *epoch* $k = 1, 2, \ldots, E_h$ **do**
  for *current batch* $b = x, y$ *from* D_i **do**
    compute loss by Eq. (9)
    $h_i^k \leftarrow h_i^{k-1} - \alpha \nabla F_h$
for *epoch* $k = 1, 2, \ldots, E_\phi$ **do**
  for *current batch* $b = x, y$ *from* D_i **do**
    $P_i^{k,(j)} \leftarrow R_i(\phi_i; x)$
    compute loss by Eq. (10) using $P_i^{k,(j)}$ and $\bar{P}^{t,(j)}$.
    $\phi_i^k \leftarrow \phi_i^{k-1} - \beta \nabla F_\phi$
$P_i^t \leftarrow \{$aggregate $P_i^{(j)}$ via Eq. (4) or Eq. (5), $j \in \mathbb{P}\}$
$\phi_i^t \leftarrow \phi_i^{E_\phi}$
Return ϕ_i^t, P_i^t

For a given activity class j, the server receives prototype sets from m clients with different data distributions. After the prototype aggregation operation, a global prototype $\bar{P}^{(j)}$ is generated for each activity class j,

$$\bar{P}^{(j)} = \frac{1}{m} \sum_{i \in m} \frac{|D_{i,j}|}{D_j} P_i^{(j)}, \tag{8}$$

where $P_i^{(j)}$ represents the prototype of activity j from client i. For example, given that the current subset contains client C_1 and client C_2, each of them has a *walking* activity prototype $P_i^{(walking)}$ computed via (4) or (5). Due to these limited feature vectors in aggregation, the *walking* prototype would pose a certain limitation. However, if the two clients upload their respective prototypes to the server, we could aggregate *walking* prototypes using (7), resulting in a global prototype $\bar{P}^{(walking)}$ with stronger representational capability.

Updating the local model parameters: During each round, m clients are selected to participate in local training. In the local training, client i first executes E_h local gradient-based updates to solve its optimal classifier given the current global representation ϕ communicated by the server. In particular, the loss function in this process is defined as follows:

$$F_h = \min_{h_i} L_i(\phi, h_i), \tag{9}$$

where $L_i(\cdot)$ is a standard cross-entropy loss for client i.

Next, the client i executes E_ϕ local updates for its representation net ϕ, while a regularization term $L_R(\cdot)$ is added to the local loss function by penalizing the distance between the local prototype $P_i^{(j)}$ and the global prototype $\bar{P}_i^{(j)}$ so as to obtain a better representation net. Specifically, for client $i (\forall i \in [N])$ the local objective of ϕ_i is to minimize the following objective function:

$$F_\phi = \min_{\phi_i} L_i(\phi_i, h_i) + \lambda L_R\left(P_i^{(j)}, \bar{P}_i^{(j)}\right), \tag{10}$$

where λ is an important hyper-parameter that controls the weight of $L_R(\cdot)$ loss. L_R can be considered a distance metric so that this function can take various forms, such as Cosine distance, L_1 distance, and L_2 distance, etc.

E. Convergence Analysis

Overall, we borrow an idea from FedProto [34] to learn a personalized HAR model for each client with convergence guarantee. Different from FedProto, besides a personalized classifier h, the local model of each client also contains the shared global feature extractor ϕ, where the overall model is parameterized by θ. As ϕ and h are highly correlated, jointly optimizing them is very difficult. Instead, we can fix h and concentrate on analyzing the convergence of ϕ, since the first step of local training is only simply updating h and not involved in too much complexity. Thus, our approach can be seen as a special case of FedProto in HAR, which still inherits its convergence properties under the relatively mild assumptions. Sharing a similar assumption to [8], [34] in deriving its convergence bound, our method can enjoy the same convergence guarantee. Upon this observation, assuming that each client's local objective function ((10)) is L_1-Lipschitz smooth bounded in $[0, G]$ and each local embedding function is L_2-Lipschitz continuous, we formulate the local convergence (i.e., *one-round deviation*) of Algorithm 1 in non-convex setting as follows:

$$\mathbb{E}\left[F_{\phi,(t+1)E_\phi+1/2}\right] \leq F_{\phi,tE_\phi+1/2} + \frac{L_1 E_\phi \beta^2}{2}\sigma^2 + \lambda L_2 \beta E_\phi G - \left(\beta - \frac{L_1\beta^2}{2}\right) \sum_{e=1/2}^{E_\phi - 1} \left\|\nabla F_{\phi,tE_\phi+e}\right\|_2^2, \tag{11}$$

which indicates the deviation bound of the local objective function for an arbitrary client after every communication round, where both $e \in \{1/2, 1, 2, \ldots, E_\phi\}$ and t denote the local iteration for ϕ and the global communication round, respectively. Here tE_ϕ represents the time step before prototype aggregation, while $tE_\phi + 1/2$ represents the time step between prototype aggregation and the first iteration of the current round. G and σ^2 denote the bounded expectation and variance of the stochastic gradient. As a result, convergence may be guaranteed because there is a certain expected one-round decrease, which could be obtained via tuning proper values for the learning rate β and the importance weight λ. A more detailed proof and analysis can be found in [34].

Authorized licensed use limited to: NANJING NORMAL UNIVERSITY. Downloaded on August 08,2023 at 01:24:53 UTC from IEEE Xplore. Restrictions apply.

TABLE I
STATISTICAL INFORMATION OF DATASETS

Dataset	PAMAP2	USC-HAD	UNIMIB-SHAR	HARBOX
Sample Rate(Hz)	33	100	50	50
Subjects	9	14	30	120
Classes	12	12	17	5
Window Size	171	512	151	900
Non-IID	✓	✓	✓	✓
Learning Rate	5e-4	1e-3	1e-2	1e-2

IV. EXPERIMENTS

A. Benchmark Datasets

We choose four public benchmark datasets for our evaluation in heterogeneous HAR scenario, which are collected by varied sensor modalities such as accelerometers and gyroscopes. To maintain the consistency with previous literature [35], the same data preprocessing such as noise filtering, normalization, and sliding window is carried out on raw sensor data. Table I summarizes the details of these datasets.

*PAMAP2 [36]:*The dataset is made up of sensor recordings obtained from nine volunteers who were asked to participate in 18 physical activities, including 12 protocol activities (*cycling*, *walking*, *rope jumping*, and so on) and a few alternative activities (*car driving*, *playing soccer*, *watching TV*, and so on). Each volunteer wore three Colibri wireless inertial measurement units(IMUs), which were attached to each dominant's chest, hand, and ankle. For further analysis, the sampling rate of 100 Hz is downsampled to 33.3 Hz.

USC-HAD [37]: This dataset is designed to serve as a benchmark for evaluating different algorithms, notably in healthcare scenarios such as elder care and health monitoring. There are 12 physical activities (*lying*, *walking*, *sitting on a chair*, etc.) collected from 14 subjects. The sampling rate is 100 Hz.

UNIMIB-SHAR [38]: This is a newly-built acceleration sensor dataset from the University of Milano Bica used to monitor human activities and detect falls. The scientists recorded the interesting activities from 30 volunteers with ages between 18 and 60 by using an Android smartphone at a sampling frequency of 50 Hz. All samples are divided into two broad categories: eight types of falls and nine types of activities of daily living (ADLs).

HARBOX [12]: This is a large-scale dataset built specifically for the FL-based HAR tasks, which includes 5 types of ADLs: *phone calls*, *walking*, *hopping*, *typing*, and *waving*, which are performed by 120 subjects with ages between 17 and 55. All sensor readings are recorded by the 9-axis IMUs embedded in 77 different brands of smartphones and resampled into 50 Hz. As indicated, HARBOX is highly heterogeneous to serve as a benchmark dataset for evaluating various FL algorithms.

B. Baselines

We compare our algorithm with eight state-of-the-art FL baselines in Non-IID setting: *SOLO:* Select all clients for local training without using FL(i.e., the computational cost is ignored); *FedAvg* [5]: The original federated averaging algorithm selects a subset of clients for local training during each round to ensure communication efficiency while sharing a global model; *FedProx* [8]: Based on FedAvg, a regularization term is added to restrain the distance between the local and global model; *MOON* [21]: The local update is corrected by maximizing the consistency between the representation learned by the local model and the representation learned by the global model; *FedProto* [34]: Based on prototype learning, a regular term is added to align features when optimizing the local model; *LG-FedAvg* [39]: Preserve a compact local representation on each client while learning a global model on all devices; *SCAFFOLD* [10]: This method corrects local updates by adding drifts to the local training; *FedHome* [15]: The global model is obtained through FedAvg, and the balanced dataset is generated to fine-tune the local model; *FedRep* [32]: Learn a shared representation and a unique local head for each client.

C. Implementation Details

Our algorithm and all the baselines are implemented using Pytorch with python 3.6, trained on a single NVIDIA GeForce RTX 3090 GPU. Unless otherwise mentioned, we randomly sample 15% clients during each round for communication efficiency. In particular, SOLO selects all clients to achieve the highest performance that local clients can attain without FL. For fair comparisons, we use the same setting on each dataset for all experiments. We train overall 300 communication rounds for all datasets to ensure the global model can converge stably, and the local training epoch is set to 5. All methods use the same backbone classification network in Fig. 1. Models are trained using a SGD optimizer with momentum 0.9. Different datasets utilize various learning rates α (Table I) to ensure global convergence. Mini-batch size B is 32 for all methods, and personalized learning rate β is 10^{-2} in local updating. For the regularization terms in the partial baselines, we utilize the default hyper-parameters from the original articles, from which we select the regularization term coefficient with the best performance on four HAR datasets. We implement the publicly released code for all baselines. Meanwhile, our code is available at[1]

In fact, most existing HAR benchmark datasets have been mainly collected in a controlled situation without significant dynamics, which actually do not suffer from heterogeneity in both label and signal distribution. This is not consistent with the real-world scenario, in which the clients might be highly diverse. Without heterogeneity, FL would always lead to a close performance to a centrally trained model. How to construct data heterogeneity in both label and signal distribution remains an important challenge in FL-based HAR. To address this issue, Li et al. [33] have empirically verified the presence of heterogeneity across clients even for the same activity. In particular, following the same strategy introduced in [4], [33], we randomly remove a subset of activities (i.e., two activity classes) for each client from its local dataset so as to mimic the real-world scenario, which forces these public datasets to follow a Non-IID distribution among clients. Thereafter, these public datasets would have heterogeneity according to both label and signal distribution. It is worth noting that our main goal is to learn a personalized model for each client, which can solve local HAR problem. In such heterogeneous FL setting, referring to previous literatures [4],

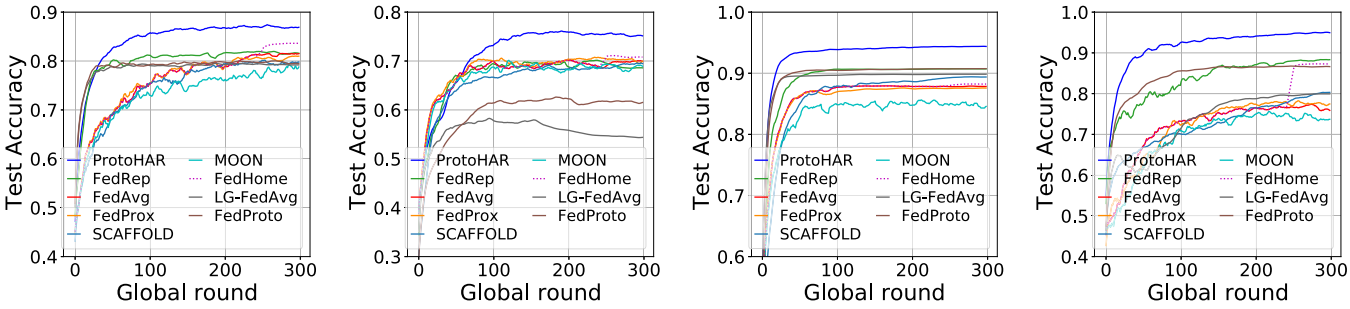
[1][Online]. Available: https://github.com/cheng-haha/ProtoHAR.

Authorized licensed use limited to: NANJING NORMAL UNIVERSITY. Downloaded on August 08,2023 at 01:24:53 UTC from IEEE Xplore. Restrictions apply.

TABLE II
MAIN RESULTS

Method	PAMAP2			USC-HAD			UNIMIB-SHAR			HARBOX		
	Acc	F1	AUC	Acc	F1	AUC	Acc	F1	AUC	Acc	F1	AUC
SOLO	80.023	79.471	95.154	55.772	50.063	84.099	90.091	86.978	98.431	79.326	76.763	91.805
FedAvg	81.867	72.104	93.362	70.812	59.188	90.633	87.968	83.158	97.950	77.621	62.399	91.004
FedProx	81.335	71.906	93.232	71.320	59.993	90.090	87.625	83.263	97.903	78.656	63.799	91.150
MOON	80.280	70.064	94.184	70.855	59.939	91.381	86.276	80.929	97.752	76.404	61.359	91.285
FedProto	80.179	79.688	96.093	62.744	57.887	92.356	90.767	87.360	98.936	86.810	85.419	96.292
SCAFFOLD	80.955	72.461	94.505	69.842	59.374	85.004	89.439	85.569	98.201	80.427	68.240	92.302
LG-FedAvg	79.643	79.097	95.347	58.491	53.017	84.635	89.846	86.766	98.546	79.998	77.730	92.185
FedHome	83.620	79.276	93.765	71.149	61.096	90.633	88.260	83.308	98.008	87.319	79.905	95.354
FedRep	82.110	81.742	94.279	70.582	65.842	90.401	90.735	87.781	98.239	88.491	88.206	95.075
ProtoHAR	**87.727**	**87.336**	**97.839**	**76.416**	**71.714**	**96.518**	**94.470**	**92.088**	**99.678**	**95.110**	**95.034**	**99.086**

The bold font highlights the best results.



(a) PAMAP2 (b) USC-HAD (c) UNIMIB-SHAR (d) HARBOX

Fig. 3. Test accuracy curves trained by partial participation of all clients during each communication round.

[12], [24], each subject of the aforementioned public HAR datasets can be treated as a client with its own local data, leading to different clients. To be specific, each client's local dataset can in turn partitioned randomly into a training set (70%) and a test set (30%), used to evaluate the model performance of each client. Because different clients have heterogeneous label distribution (Non-IID or imbalance), we introduce three widely-used metrics: Macro-F1, Accuracy, and AUC (Area under the ROC Curve) for our evaluation. On this basis, the model performance is evaluated with the mean prediction metrics [33], which can be defined as follows:

$$\text{Metric} = \frac{1}{\sum_{i=1}^{N} m_i} \sum_{i=1}^{N} m_i * \mathrm{E}_i, \mathrm{E} \in \{Acc, F1, AUC\}, \tag{12}$$

where, m_i denotes the number of test samples on client i. Overall, the evaluation methodology is more reasonable by treating each subject within public datasets a client with its own local dataset.

TABLE III
THE COMMUNICATION ROUNDS IN DIFFERENT METHODS TO ACHIEVE THE SAME TARGET ACCURACY

Method	PAMAP2		USC-HAD		UNIMIB-SHAR	
	#R	S ↑	#R	S ↑	#R	S ↑
#R and S ↑ for FedAvg						
FedAvg	280	1.00 ×	212	1.00 ×	196	1.00 ×
FedProx	\	\	89	2.38 ×	\	\
MOON	\	\	128	1.66 ×	\	\
FedProto	\	\	\	\	26	7.54 ×
SCAFFOLD	\	\	\	\	87	2.25 ×
LG-FedAvg	\	\	\	\	26	7.54 ×
FedHome	252	1.11 ×	253	0.84 ×	255	0.77 ×
FedRep	246	1.14 ×	\	\	40	4.9 ×
ProtoHAR	**53**	**5.28 ×**	**78**	**2.72 ×**	**20**	**9.8 ×**
#R and S ↑ for FedRep						
FedRep	249	1.00 ×	150	1.00 ×	260	1.00 ×
ProtoHAR	**53**	**4.7 ×**	**76**	**1.97 ×**	**26**	**10.0 ×**

In addition, we denote the communication round of each method to achieve the target accuracy as "#R", the corresponding convergence speedup relative to the target method as "S ↑".
The bold font highlights our results.

D. Comparison With State-of-The-Art Methods

Table II reports the quantitative results of our algorithm and current state-of-the-art baselines on four benchmark datasets in terms of three metrics. Initially, we experiment on three random seeds (i.e. 0,1,2) and average the results. Overall, the proposed ProtoHAR method outperforms all baselines in all metrics, demonstrating that our model can effectively address data heterogeneity in HAR. We attribute this to the transferred knowledge with enhanced generalization capability by correcting global prototypes in each local representation. Under such heterogeneous setting, SOLO (local training without FL) shows much worse accuracy than partial FL baselines, leading to relatively low accuracy for all clients. This suggests the necessity of FL in HAR. In addition, the lower performance caused

Authorized licensed use limited to: NANJING NORMAL UNIVERSITY. Downloaded on August 08,2023 at 01:24:53 UTC from IEEE Xplore. Restrictions apply.

by traditional FL baselines proves that they do not solve the heterogeneous HAR problem, which indicates the advantage of our ProtoHAR. Table III compares their convergence speeds on three public HAR datasets and Fig. 3 presents the test accuracy curves when training all FL algorithms. We further detail them on each dataset:

Evaluation on PAMAP2: 1) Since PAMAP2 inherently contains unbalanced data distribution among clients, traditional FL baselines suffer from data heterogeneity and do not converge well. Our method outperforms the best baseline, i.e., FedHome (above 4.107%, 8.06%, and 4.074%) in the three metrics. 2) As one can see from Fig. 3(a), ProtoHAR yields the most rapid learning curves to achieve the desired performance and outperforms other baselines, e.g., as shown in Table III, ProtoHAR requires only 53 communication rounds to achieve the best accuracy that FedAvg takes 280 rounds to reach. That is, ProtoHAR converges nearly 5.28× faster than FedAvg. Similarly, it is 4.7× faster than FedRep.

Evaluation on UNIMIB-SHAR: 1) Regarding UNIMIB-SHAR, SOLO has a higher metric score. We attribute this to the fact that each client's local data is sufficient to learn partial activity classes well. In contrast, FedAvg has more complex optimization objectives, and the aggregated global model fails to address such data heterogeneity among clients. Hence it exhibits worse performance. As shown in Fig. 3(c), our algorithm performs the best and significantly outperforms SOLO, which we attribute to the reason that the prototypes can better guide the local training to learn more knowledge. 2) Our method achieves a remarkable performance improvement (over 3.703%, 4.728% and 0.742%) in three metrics and greatly exceeds FedAvg, FedProx, and MOON in F1-score (over 8%).

Evaluation on USC-HAD: 1) It can be found that LG-FedAvg shows a clear decline trend in accuracy. Though sharing a single global classifier leads to a smaller memory footprint, we argue that it could not alleviate the overfitting problem well. 2) Regarding other baselines, FedProx achieves the best Accuracy (i.e., 71.320%), while FedRep obtains the best Macro-F1 score (i.e., 65.842%), and MOON obtains the best AUC performance (i.e., 91.381%). These methods do not yield a consistent improvement across all metrics. In comparison, our method produces higher metric scores and achieves an Accuracy of 76.416% (5.096% gain), a Macro-F1 of 71.714% (5.872% gain), and an AUC of 96.518% (5.137% gain). 3) Table III shows that our method can lead to an acceleration of convergence by 2.72× and 1.97× compared with FedAvg and FedRep, respectively. 4) We notice that Tan et al. [34] have recently proposed the concept of prototype learning to address data heterogeneity in such FL setting. Though their FedProto has claimed decent performance in multiple image classification tasks [34], it still remains a question whether it can be directly applied in HAR area. Unlike image data, besides Non-IID label distribution in HAR environment, different clients might have heterogeneity in signal distribution even for the same activity. Inspired by above analyses, the feature alignment is first embedded into the framework of prototype learning. We compare the FedProto with our method in the same experimental setting. As shown in Table II, it can be observed that only sharing global prototypes without the shared global feature extractor is not enough and even causes substantial performance degradation, which indicates the FedProto alone does not work well for the HAR problem. For example, one can clearly observe an accuracy reduction of FedProto compared to our approach from over 76% to below 63% on USC-HAD. The experimental results verify the contribution in the part of feature alignment, which are consistent with our intuition so that we can better understand why the feature alignment does make sense here for HAR problem due to the heterogeneity in signal distribution.

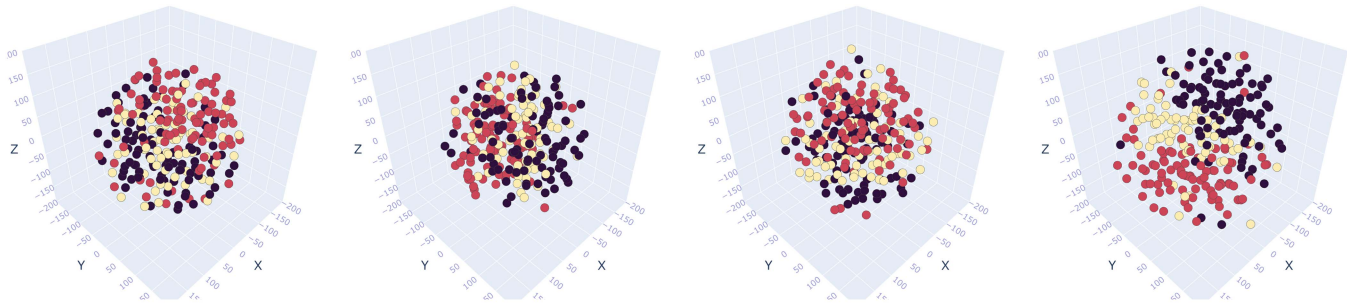
Evaluation on HARBOX: 1) Our method also outperforms the best baseline, FedRep (above 6.619%, 6.828%, and 4.011%) in the three metrics. It can be observed that FedHome achieves the second-best result among all baselines. However, it is still based on a single global model, which does not consider the data heterogeneity problem among clients at an early stage during the FL process. 2) HARBOX is a large-scale dataset containing 120 users, where a relatively small number of activity classes leads to more severe data heterogeneity. As Table II shows: the traditional FL algorithms do not bring a significant accuracy improvement to the clients, while our algorithm achieves the best performance due to the flowing activity prototypes among clients, which empirically validates that our approach can improve personalization for each client in a large-scale scenario.

Visualization Analysis: T-SNE is utilized to visualize three fine-grained activity categories (*walking forward*, *walking left*, *walking right*) on the USC-HAD test set in a 3-dimensional space. As shown in Fig. 4, it can be seen that the feature representations learned by SOLO, FedAvg, and FedRep are difficult to distinguish, which results in a loss of generalization ability on heterogeneous clients. Our method shows a clear separation, which verifies our motivations that our ProtoHAR could benefit from the global prototype knowledge flowing among local clients, which alleviates the disparity of underlying data distribution among clients. This knowledge is otherwise not accessible through other baselines such as FedAvg or FedRep. Moreover, comparing to Fig. 4(c) and (d) visually demonstrates that modifying local representation based on prototypes is more beneficial for learning client-shared activity features than directly embedding all activity instances from heterogeneous clients into a common representation space. Based on the fact that ProtoHAR effectively alleviates the insufficiency of local activity knowledge by embedded global activity prototypes, it can efficiently optimize the objectives and result in accelerated convergence speed, while requiring at most one-half as overall communication rounds as FedAvg. In particular, as aforementioned, our approach also saves transmission cost and computation overhead, which proves that it can well address the data heterogeneity problem and enable fast personalization for each client in HAR scenario.

E. Ablation Studies

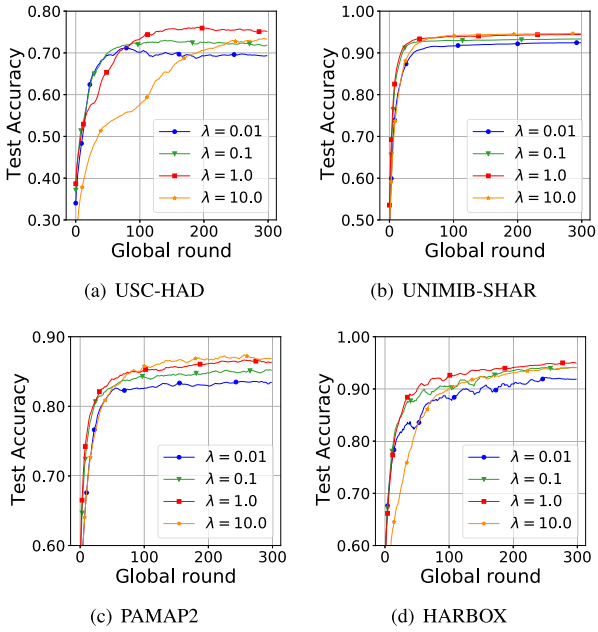
Effects of major components: To prove the effectiveness of PCR (**P**rototypes **C**orrects local **R**epresentation), we use *W.O.PCR* to denote the variant that does not use prototypes to correct representation. To demonstrate the effectiveness of SCR (**S**olve for the optimal **C**lassifier on global **R**epresentation), we use *W.O.SCR* to denote the variant that does not share the global representation but share the global model. On USC-HAD, Fig. 6 shows that both *W.O.PCR* and *W.O.SCR* perform worse than

Authorized licensed use limited to: NANJING NORMAL UNIVERSITY. Downloaded on August 08,2023 at 01:24:53 UTC from IEEE Xplore. Restrictions apply.



(a) Local model (b) FedAvg (c) FedRep (d) ProtoHAR

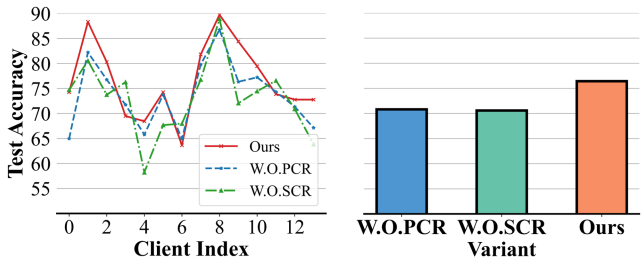
Fig. 4. T-SNE visualizations of representation vectors. Different colors represent different categories.



(a) USC-HAD (b) UNIMIB-SHAR

(c) PAMAP2 (d) HARBOX

Fig. 5. Effection of λ.



(a) Accuracy on each client (b) Average accuracy for different variants

Fig. 6. Ablation experiments on different variants.

ours across most clients, which suggests that our method greatly boosts the performance for most clients, hence validating our motivation to leverage prototypes to guide local representation in personalized training.

Effects of hyper-parameters: In this part, we evaluate the two main hyper-parameters, λ and E_ϕ/E_h, in ProtoHAR. Fig. 5

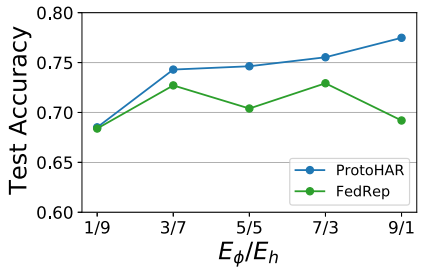


Fig. 7. Effection of alternating update ratio.

shows the convergence rate of ProtoHAR trained on four HAR datasets with different λ. It can be seen that too large λ will hurt the performance of ProtoHAR, which causes ProtoHAR to diverge. Therefore, λ should be properly tuned. We set $\lambda = 10.0, 1.0, 1.0, 1.0$ for PAMAP2, UNIMIB-SHAR, USC-HAD, and HARBOX in all scenarios, respectively. As alternating update ratios E_ϕ/E_h represents the level of corrected representation ϕ, E_ϕ/E_h can also be considered as a crucial hyper-parameter in ProtoHAR. In Fig. 7, fixing the number of local epochs at 10, we analyze the impact of different ratios on performance. The increase in the number of E_ϕ indicates that the network has stronger feature extraction capability, which may lead to accuracy improvements in both cases. Our method achieves the best results across all ratios, compared with FedRep, which adopts an alternating update strategy as well. Our algorithm performs better as E_ϕ increases, which proves that the global representation obtained with more prototype correction is more informative. For fair comparisons, we choose $E_\phi/E_h = 7/3$ with the best performance in FedRep as our default setting. If higher performance is desired, ProtoHAR can utilize $9/1$ to adjust the local training dynamically.

Effects of different strategies for aggregating prototypes: We assume that aggregating prototypes on sparse samples are inadequate and propose reweighted prototypes using (5) to amend the bias. As shown in Table IV, we can let ProtoHAR (+ RP) get a slight rise in Accuracy and Macro-F1 by adjusting the scale parameter ϵ. The results show that reweighting strategy yields a better prototype to guide local training for USC-HAD dataset. Although the performance boost is not significant for PAMAP2 dataset, it just proves that aggregating global prototypes can help to eliminate differences between local prototypes on clients. That is to say, when a certain type of prototype lacks

Authorized licensed use limited to: NANJING NORMAL UNIVERSITY. Downloaded on August 08,2023 at 01:24:53 UTC from IEEE Xplore. Restrictions apply.

TABLE IV
ABLATION STUDY OF OUR REWEIGHTED PROTOTYPE METHOD (+ RP) IN TERMS OF DIFFERENT ϵ AGAINST PROTOHAR

Methods	e	USC-HAD		PAMAP2	
		Acc	F1	Acc	F1
ProtoHAR	\	76.416	71.714	87.727	87.336
ProtoHAR + RP	0.01	80.813	76.531	87.876	87.732
	0.1	80.716	76.938	87.541	87.414
	1.0	80.412	76.736	87.490	87.391
	10.0	80.816	76.703	87.829	87.670

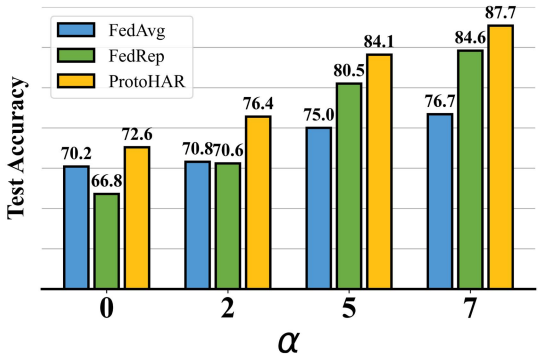


Fig. 8. Performance on different data splits.

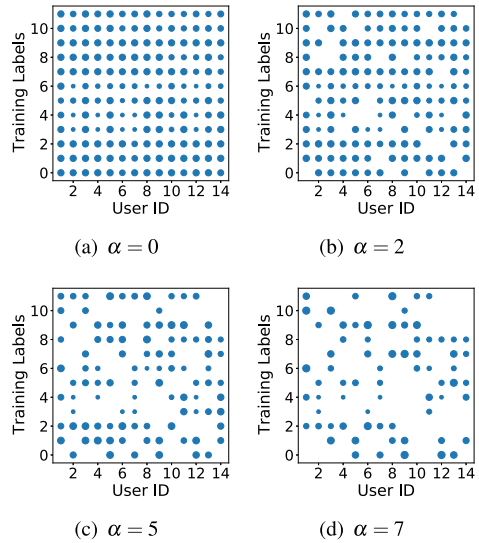


Fig. 9. Effection of α.

enough knowledge, the prototype can bridge the gap through aggregating the more informative prototypes from other clients.

F. Robustness Analysis

Robustness on heterogeneous data: A more heterogeneous or imbalanced data would significantly slow down model convergence [11]. Fig. 9 shows different pathological splits: $\alpha = 0, 2, 5, 7$. The α indicates the number of deleted categories. It can be seen that when $\alpha = 0$, the data distribution of USC-HAD is IID, while the larger α, the larger the difference between clients,

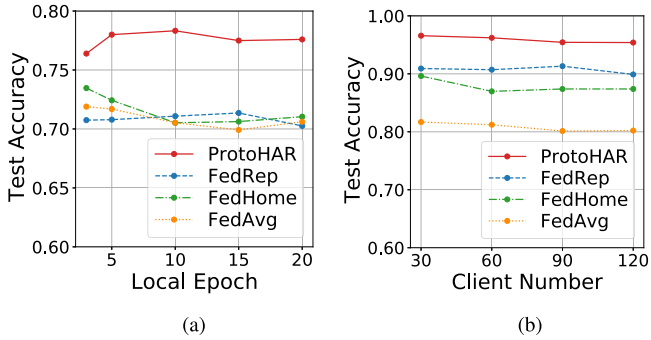


Fig. 10. Robustness analysis.

i.e., stronger heterogeneity. As shown in Fig. 8, it can be found that our method maintains strong robustness against different data splittings, the more heterogeneous the data distribution, the better the personalization of our approach, which may be because the unique classifier trained on the prototype-guided representation can better adapt to the local distribution. On the other hand, when the data is IID ($\alpha = 0$), the accuracy of the personalization method FedRep decreases while that of FedAvg that is sensitive to heterogeneity increases, and ProtoHAR is superior to these two methods, which indicates that our method can strike a better balance between personalization and generalization.

Robustness on different local epochs: Fig. 10(a) shows the effect of the number of local update epochs on USC-HAD. As the number of local epochs increases, our approach outperforms several competitive baselines. We attribute this to an enhanced generalization ability of prototype-guided local representations, where the local classifier trained on the aggregated global representation can better recognize potential target activities.

Robustness on different client numbers: We conduct experiments to analyze the robustness of ProtoHAR by limiting different numbers of clients from the HARBOX dataset to participate in the training process. We report the model accuracy in Fig. 10(b). Our method consistently achieves the best performance, which implies that it can be applicable in a large-scale HAR scenario.

V. CONCLUSION

FL has substantial obstacles to HAR due to the presentence data heterogeneity in label and signal distribution among clients. In this article, we propose a novel FL framework named ProtoHAR, which aims to decouple representations and classifiers to mitigate the above problem more efficiently and effectively. ProtoHAR leverages the global activity prototype knowledge flowing among different clients to correct local representation. Based on the improved representation, the user-specific classifier is optimized to make it more discriminative for personalized HAR. Extensive experiments on four public HAR datasets have demonstrated the effectiveness of the proposed framework in both classification performance and communication efficiency. ProtoHAR can be applied to many real-world HAR scenarios to build accurate personalized activity classification models for mobile users without collecting raw sensor data in a central

Authorized licensed use limited to: NANJING NORMAL UNIVERSITY. Downloaded on August 08,2023 at 01:24:53 UTC from IEEE Xplore. Restrictions apply.

server. In essence, the proposed ProtoHAR is a stationary solution, which assumes that the activity data distribution always remains the same or unchanged. In practice, activity data usually tends to flow and increase in terms of size over time. Thus, it is essential to refine the prediction model since deep neural networks have to continuously learn new activity data in a non-stationary scenario [20]. However, ProtoHAR is not able to continuously learn new activity data, which needs to be retrained from scratch. This is unrealistic for deep models because of the high cost during training process. On the other hand, the continuous learning can help to dynamically learn new knowledge from new activity instances, which could properly address the changes in data distribution. To our knowledge, the concept of prototypes has not been fully considered in a non-stationary scenario. In a future study, we plan to extend the concept of prototypes from another perspective of continuous learning, which handles a continuous data flow to each client with a non-stationary distribution in such FL setting, so as to provide a better real-time performance in practical HAR applications.

REFERENCES

[1] A. Bulling, U. Blanke, and B. Schiele, "A tutorial on human activity recognition using body-worn inertial sensors," *ACM Comput. Surveys*, vol. 46, no. 3, pp. 1–33, 2014.

[2] K. Chen, D. Zhang, L. Yao, B. Guo, Z. Yu, and Y. Liu, "Deep learning for sensor-based human activity recognition: Overview, challenges, and opportunities," *ACM Comput. Surveys*, vol. 54, no. 4, pp. 1–40, 2021.

[3] Q. Yang, Y. Liu, T. Chen, and Y. Tong, "Federated machine learning: Concept and applications," *ACM Trans. Intell. Syst. Technol.*, vol. 10, no. 2, pp. 1–19, 2019.

[4] K. Sozinov, V. Vlassov, and S. Girdzijauskas, "Human activity recognition using federated learning," in *Proc. ISPA/IUCC/BDCloud/SocialCom/SustainCom*, 2018, pp. 1103–1111.

[5] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proc. Artif. Intell. Statist.*, 2017, pp. 1273–1282.

[6] P. Kairouz et al., "Advances and open problems in federated learning," *Found. Trends Mach. Learn.*, vol. 14, no. 1–2, pp. 1–210, 2021.

[7] Q. Li, Y. Diao, Q. Chen, and B. He, "Federated learning on non-IID data silos: An experimental study," 2021, *arXiv:2102.02079*.

[8] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," *Proc. Mach. Learn. Syst.*, vol. 2, pp. 429–450, 2020.

[9] Y. Zhao, M. Li, L. Lai, N. Suda, D. Civin, and V. Chandra, "Federated learning with non-IID data," 2018, *arXiv:1806.00582*.

[10] S. P. Karimireddy, S. Kale, M. Mohri, S. Reddi, S. Stich, and A. T. Suresh, "Scaffold: Stochastic controlled averaging for federated learning," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 5132–5143.

[11] X. Li, K. Huang, W. Yang, S. Wang, and Z. Zhang, "On the convergence of FEDAVG on non-IID data," 2019, *arXiv:1907.02189*.

[12] X. Ouyang, Z. Xie, J. Zhou, J. Huang, and G. Xing, "Clusterfl: A similarity-aware federated learning system for human activity recognition," in *Proc. 19th Annu. Int. Conf. Mobile Syst., Appl., Serv.*, 2021, pp. 54–66.

[13] R. Presotto, G. Civitarese, and C. Bettini, "Fedclar: Federated clustering for personalized sensor-based human activity recognition," in *Proc. IEEE Int. Conf. Pervasive Comput. Commun.*, 2022, pp. 227–236.

[14] Y. Chen, X. Qin, J. Wang, C. Yu, and W. Gao, "Fedhealth: A federated transfer learning framework for wearable healthcare," *IEEE Intell. Syst.*, vol. 35, no. 4, pp. 83–93, Jul./Aug. 2020.

[15] Q. Wu, X. Chen, Z. Zhou, and J. Zhang, "Fedhome: Cloud-edge based personalized federated learning for in-home health monitoring," *IEEE Trans. Mobile Comput.*, vol. 21, no. 8, pp. 2818–2832, Aug. 2022.

[16] J. Snell, K. Swersky, and R. Zemel, "Prototypical networks for few-shot learning," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017.

[17] G. Paragliola, "Evaluation of the trade-off between performance and communication costs in federated learning scenario," *Future Gener. Comput. Syst.*, vol. 136, pp. 282–293, 2022.

[18] Y. Zhao, M. Li, L. Lai, N. Suda, D. Civin, and V. Chandra, "Federated learning with non-iid data," 2018, *arXiv:1806.00582*.

[19] M. Duan et al., "Astraea: Self-balancing federated learning for improving classification accuracy of mobile deep learning applications," in *Proc. IEEE 37th Int. Conf. Comput. Des.*, 2019, pp. 246–254.

[20] G. Paragliola, "A federated learning-based approach to recognize subjects at a high risk of hypertension in a non-stationary scenario," *Inf. Sci.*, vol. 622, pp. 16–33, 2023.

[21] Q. Li, B. He, and D. Song, "Model-contrastive federated learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 10713–10722.

[22] F. J. Ordóñez and D. Roggen, "Deep convolutional and LSTM recurrent neural networks for multimodal wearable activity recognition," *Sensors*, vol. 16, no. 1, 2016, Art. no. 115.

[23] J. Yang, M. N. Nguyen, P. P. San, X. L. Li, and S. Krishnaswamy, "Deep convolutional neural networks on multichannel time series for human activity recognition," in *Proc. 24th Int. Joint Conf. Artif. Intell.*, 2015, pp. 3995–4001.

[24] S. EK, F. Portet, P. Lalanda, and G. Vega, "A federated learning aggregation algorithm for pervasive computing: Evaluation and comparison," in *Proc. IEEE Int. Conf. Pervasive Comput. Commun.*, 2021, pp. 1–10.

[25] Z. Xiao, X. Xu, H. Xing, F. Song, X. Wang, and B. Zhao, "A federated learning system with enhanced feature extraction for human activity recognition," *Knowl.-Based Syst.*, vol. 229, 2021, Art. no. 107338.

[26] G. M. Weiss and J. Lockhart, "The impact of personalization on smartphone-based activity recognition," in *Proc. Workshops 26th AAAI Conf. Artif. Intell.*, 2012.

[27] J. Liu, L. Song, and Y. Qin, "Prototype rectification for few-shot learning," in *Proc. 16th Eur. Conf. Comput. Vis.*, 2020, pp. 741–756.

[28] B. Yang, C. Liu, B. Li, J. Jiao, and Q. Ye, "Prototype mixture models for few-shot semantic segmentation," in *Proc. 16th Eur. Conf. Comput. Vis.*, 2020, pp. 763–778.

[29] Y. Pan, T. Yao, Y. Li, Y. Wang, C.-W. Ngo, and T. Mei, "Transferrable prototypical networks for unsupervised domain adaptation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 2239–2247.

[30] S. Caldas et al., "Leaf: A benchmark for federated settings," 2018, *arXiv:1812.01097*.

[31] F. Cruciani et al., "Personalizing activity recognition with a clustering based semi-population approach," *IEEE Access*, vol. 8, pp. 207794–207804, 2020.

[32] L. Collins, H. Hassani, A. Mokhtari, and S. Shakkottai, "Exploiting shared representations for personalized federated learning," in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 2089–2099.

[33] C. Li, D. Niu, B. Jiang, X. Zuo, and J. Yang, "Meta-har: Federated representation learning for human activity recognition," in *Proc. Web Conf.*, 2021, pp. 912–922.

[34] Y. Tan et al., "Fedproto: Federated prototype learning across heterogeneous clients," in *Proc. AAAI Conf. Artif. Intell.*, vol. 1, 2022, p. 3.

[35] D. Anguita, A. Ghio, L. Oneto, X. Parra, and J. L. Reyes-Ortiz, "Human activity recognition on smartphones using a multiclass hardware-friendly support vector machine," in *Proc. Int. Workshop Ambient Assist. Living*, 2012, pp. 216–223.

[36] A. Reiss and D. Stricker, "Introducing a new benchmarked dataset for activity monitoring," in *Proc. IEEE 16th Int. Symp. Wearable Comput.*, 2012, pp. 108–109.

[37] M. Zhang and A. A. Sawchuk, "USC-had: A daily activity dataset for ubiquitous activity recognition using wearable sensors," in *Proc. ACM Conf. Ubiquitous Comput.*, 2012, pp. 1036–1043.

[38] D. Micucci, M. Mobilio, and P. Napoletano, "Unimib shar: A dataset for human activity recognition using acceleration data from smartphones," *Appl. Sci.*, vol. 7, no. 10, 2017, Art. no. 1101.

[39] P. P. Liang et al., "Think locally, act globally: Federated learning with local and global representations," 2020, *arXiv:2001.01523*.

Authorized licensed use limited to: NANJING NORMAL UNIVERSITY. Downloaded on August 08,2023 at 01:24:53 UTC from IEEE Xplore. Restrictions apply.