BalanceFL: Addressing Class Imbalance in Long-Tail Federated Learning

Xian Shuai, Yulin Shen, Siyang Jiang, Zhihe Zhao, Zhenyu Yan, Guoliang Xing*
The Chinese University of Hong Kong
{sx018,ylshen,syjiang,zz021,zyyan,glxing}@ie.cuhk.edu.hk

ABSTRACT

Federated Learning (FL) is an emerging learning paradigm that enables the collaborative learning of different nodes without exposing the raw data. However, a critical challenge faced by the current federated learning algorithms in real-world applications is the long-tailed data distribution, i.e., in both local and global views, the numbers of classes are often highly imbalanced. This would lead to poor model accuracy on some rare but vital classes, e.g., those related to safety in health and autonomous driving applications. In this paper, we propose BalanceFL, a federated learning framework that can robustly learn both common and rare classes from a long-tailed real-world dataset, addressing both the global and local data imbalance at the same time. Specifically, instead of letting nodes upload a class-drifted model trained on imbalanced private data, we design a novel local update scheme that rectifies the class imbalance, forcing the local model to behave as if it were trained on ideal uniform distributed data. To evaluate the performance of BalanceFL, we first adapt two public datasets to the long-tailed federated learning setting, and then collect a real-life IMU dataset for action recognition, which includes more than 10,000 data samples and naturally exhibits the global long tail effect and the local imbalance. On all of these three datasets, BalanceFL outperforms state-of-the-art federated learning approaches by a large margin.

KEYWORDS

federated learning, long-tailed learning

1 INTRODUCTION

Federated learning is a promising approach for Internet of Things (IoT) as it enables model training on decentralized data residing on local nodes. It not only unleashes the compute power of the edge but also takes advantage of the distributed data for collaborative learning. A typical federated learning approach (e.g., FedAvg) [31] aggregates the model weights from all nodes iteratively until converging to a global model. As only model weights are required to upload, it avoids exposing users' raw data during the learning process. However, existing federated algorithms face several major challenges in real-world IoT applications, as illustrated in Figure 1.

The first challenge is the *global data imbalance*. The overall data, i.e., the union of all distributed data, may follow a long-tailed distribution. Specifically, there may exist both head classes attaching with a large amount of data, and tail classes that are rare and only have a small number of data samples. The direct impact of class imbalance is a drift to head classes and the performance drop of classification accuracy on minority tail classes. However, those minority classes may play a much more important role beyond

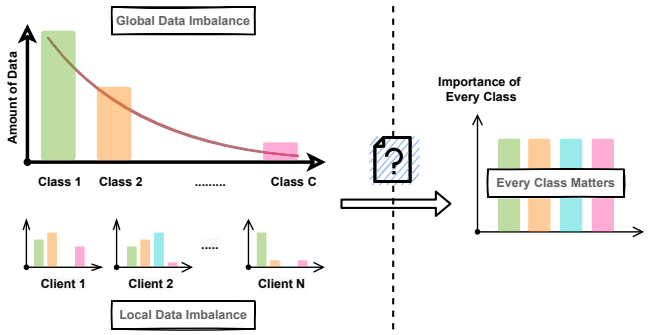


Figure 1: The illustration of two practical challenges: global data imbalance and local data imbalance. Our goal is to maximize the overall accuracy of all classes with the imbalanced training data under the federated setting.

their proportion in data, especially in critical applications related to safety or health, such as autonomous driving and medical diagnosis [2, 38]. Recently, a number of centralized learning approaches have been proposed to address this long-tailed distribution problem [3, 8, 18, 32, 45]. A common approach is to add a compensation term to either the loss or the prediction results for the tail classes, based on the training data distribution. However, such an approach is not applicable in federated learning because only weights or gradients of the model can be exchanged between nodes and the server, due to the privacy concern. As a result, the global data distribution is unobservable for both nodes and the server, making the adoption of these centralized re-balancing approaches infeasible. Until recently, only few studies investigated the long-tailed federated learning problem. Astraea [11] addresses this challenge by requiring nodes to upload the number of local samples of each class to the server. However, it exposes a latent backdoor to attackers and can cause privacy leakage. In [48], the global distribution is inferred by observing the gradient changes corresponding to every class. However, it requires an additional validation dataset on the server, and relies on the strong assumption that there exists a correlation between gradient magnitude and sample quantity.

In addition to global imbalance, another major challenge is *local data imbalance*, i.e., the sample number of every class on a node is highly uneven. An extreme case of the local data imbalance is the class missing, where some classes are not present at all while the node is still required to recognize them. The class missing issue is particularly likely to occur for those global tail classes, as tail classes are rare in terms of both quantity and occurrence frequency. There may also exist cases where a certain class is the majority class on some nodes but the minority class on other nodes, which results

*Corresponding Author.

978-1-6654-9624-7/22/$31.00 ©2022 IEEE
DOI 10.1109/IPSN54338.2022.00029

in the inter-node data heterogeneity, also known as the non-iid (independent identically distributed) data. An emerging paradigm to address the data heterogeneity is the personalized federated learning [43, 44, 55]. However, these approaches do not consider the global data imbalance or the distribution mismatch between the local training data and the test data.

In this paper, we propose BalanceFL, a novel long-tail federated learning framework that can handle the global and local data imbalance simultaneously in a unified manner. Instead of letting nodes upload biased local models trained on imbalanced private data,we design a new local self-balancing scheme, which forces the uploaded local model to behave as if it were trained from a uniform distribution dataset. This design is motivated by two key insights. First, from the perspective of data, as long as the "pseudo local data" was balanced for every node, the overall global data would also be "pseudo-balanced", which avoids the difficulty in estimating the global data distribution, and would allow the algorithms designed for global balanced data (e.g., FedAvg) to perform well. In Section 4.1, we provide a theoretical foundation for this intuition. Second, by applying the local self-balancing, the side-effect of data heterogeneity among nodes can be alleviated, since all local models are regularized to resemble the model trained from an evenly distributed dataset, leading to faster convergence. In order to achieve the above self-balancing, we disassemble the issue of local data imbalance into two sub-problems: class missing and the data amount imbalance on those classes that have data. In BalanceFL, we address them respectively by two techniques: knowledge inheritance and inter-class balancing. Different from existing works [11, 48], our approach works with existing federated learning settings, without imposing any additional requirements such as the uploading of local data distribution or an additional dataset on the server.

To validate the performance of BalanceFL, we first evaluate it on two constructed datasets, the long-tailed version of CIFAR10 [24] and Google Speech Commands [51], whose modalities are image and audio, respectively. In addition, we collect a real-life IMU dataset for action recognition, in which we observe a natural long tail effect. Results show that BalanceFL largely outperforms other state-of-the-art federated learning algorithms. Specifically, it improves the overall accuracy of FedAvg [31] by up to 56.7% , 39.5%, 18.5% on three datasets, respectively. Moreover, it achieves 75%, 33.6%, 68.6% less communication overhead compared with FedAvg.

In summary, we conclude our contributions as follows:

- We formally identify two types of class imbalance in federated learning: local imbalance and global imbalance. We use a case study to illustrate the issue and present theoretical foundation to motivate our approach.
- We propose BalanceFL, a novel long-tail federated learning framework addressing both global and local imbalance. To the best of our knowledge, this is the first framework that enables federated learning algorithms to achieve satisfactory performance under long-tailed datasets.
- We collect a real-life IMU dataset for action recognition, which includes over 10,000 data samples. It naturally exhibits a global long tail effect and local imbalance. The dataset and our implementation of BalanceFL as well as six baselines are made available to the community.[1]
- We conduct extensive experiments on three datasets of different modalities. All experiments validate the superior performance of BalanceFL over existing state-of-the-art federated learning algorithms.

The rest of this paper is organized as follows. Section 2 introduces the related work. Section 3 presents the background. Section 4 presents a motivation study. Section 5 describes the system overview and the design. In the next, we show how we generate the dataset in Section 6, and evaluate BalanceFL in Section 7. Finally, in Section 8, we conclude the paper and discuss the future works.

2 RELATED WORK

2.1 Personalized Federated Learning

An emerging paradigm in federated learning is personalized federated learning [44], which has two key features. First, by leveraging a corpus of decentralized data residing on different devices, the personalized federated learning alleviates the over-fitting of the local training and improves the generalization ability. Second, compared with FedAvg, personalized federated learning can better handle the data heterogeneity among nodes. Specifically, in [55], federated learning is combined with the meta-learning algorithm MAML [13], enabling the global learned model to quickly adapt to the local data for personalization. Dinh et al. [43] propose to use Moreau envelops as the regularized loss function to decouple the personalized model optimization from the global model learning. Other studies [33, 40] resort to multi-task learning, which treat the model personalization for every node as a different task and perform the joint optimization. There are also works associating personalized models with the global model. Chen et al. [4, 53] show that fine-tuning the global model learned from a generic federated learning algorithm such as FedAvg already achieves satisfactory personalized performance. GRP-FED [20] jointly utilizes a global model that is obtained from adaptive aggregation to ensure fairness among clients, and local models to customize each client. Although personalized federated learning can handle data heterogeneity by adapting to each client's local data, it assumes certain similarity between the training and test data distributions and cannot address the global imbalance. In our setting, the local training data is non-ideal and imbalanced, where there may be only few or even no samples on some classes that are of interest. This puts forward demands on new algorithms that are specifically designed for the nature of imbalance.

2.2 Long-Tail Learning

As a result of the prevalence of the imbalanced data distribution in real-world applications, long-tail learning has attracted significant attention. Several solutions have been proposed, including re-sampling, loss re-weighting and knowledge transfer from head to tail classes. Re-sampling based approaches over-sample the minority classes or under-sample the majority classes to calibrate the imbalanced distribution [1, 10]. Loss re-weighting based approaches add a class-specific compensation term to the original loss function to re-balance the contribution of every class [8, 18, 36, 45]

[1] https://github.com/sxontheway/BalanceFL

Other works aim to enhance the representation learning of the tail classes by transferring knowledge from head classes [6, 29, 49]. However, these methods usually require a specific design of the neural network structure, which restricts the generalizability. Apart from above approaches, there are also several recent works [22, 56] that aim to improve the long-tail prediction by decoupling the representation learning and the training of the classifier into two stages. Although long-tail learning has been widely studied in the centralized setting, it is difficult to directly adapt those solutions to federated setting, because the global training distribution is unknown and can be significantly different from the local training distribution on each client.

3 BACKGROUND

3.1 Application Scenarios

BalanceFL is applicable to a wide range of scenarios where distributed nodes intend to collaboratively learn a deep learning model, while the overall data exhibits skewed distributions, rather than the ideal uniform distributions over each class. The Long tail is a very common phenomenon in an open world. Zipf's law [35] in linguistics indicates that a few words are used a lot while a lot of words are used infrequently. Similarly, the frequency distributions of categories of both visual and the biological data also exhibit the long tail effect [19, 46].

Representative applications of BalanceFL include smart home and healthcare monitoring systems. These systems are typically designed to recognize a variety of users' behaviors including speech commands and indoor/outdoor activities, using wearable or mounted sensors [37, 47]. However, since many events only occur sporadically, an individual user may even not have any samples of certain classes, let alone train a deep learning model that can recognize them. A straightforward solution to tackle this local-side class missing issue is to aggregate data from all users. However, the sensing data for healthcare and smart homes are privacy-sensitive and hence cannot be shared or uploaded. In addition, even if all local data are aggregated together into a global dataset, it may still suffer a long-tail effect. For example, in smart homes scenarios, common activities such as sitting, lying, and walking account for the majority of data for all families. Samples of classes such as falling down or stomping, which are vital to human safety or are important digital biomarkers for chronic diseases diagnosis such as early Alzheimer's [37], are rare for most families.

Basically, there exist two federated learning settings [21]. In the cross-device setting, there is usually a huge number of clients where each client could be stateless and is likely to appear only once during the whole training. In cross-silo federated learning, the number of silos is limited, and each silo (e.g., a hospital) can participate in the training continuously. BalanceFL mainly focuses on addressing the global class imbalance issue when the number of clients is relatively small and each client can participate in the training for multiple times, which is similar to the cross-silo setting. However, we also note that BalanceFL can work under partial participation (see Section 7.4), which is one of the critical characteristics of cross-device federated learning.

3.2 Problem Formulation

Local and Global Data Distribution. Assume the final objective of federated learning is to train a global model, which is able to recognize C classes, and the total number of nodes involved in federated learning is N. We use m_j^i to denote the number of sample of class j owned by node i, where $i \in \{1, .., N\}$ and $j \in \{1, ..., C\}$.[2] Then we can denote the (unnormalized) local data distribution on node i by: $d_{local}^i = \{m_1^i ..., m_C^i\} \in \mathbb{R}^C$. The global data distribution can be obtained by adding the numbers of samples of each class on each node together: $d_{global} = \{\sum_{i=1}^{N} m_1^i, ..., \sum_{i=1}^{N} m_C^i\} \in \mathbb{R}^C$. Locally, when some elements in d_{local}^i are zero, the *class missing* issue occurs on node i. From the global perspective, the *global imbalance* happens when elements in d_{global} are not equal to each other. Specifically, we can define the global imbalance ratio as the quotient between the total sample number of the majority class across all nodes and that of the minority class, i.e., $\Gamma = max(d_{global})/min(d_{global})$. Γ will be used in Section 6.1.1 for the experimental dataset generation.
Classification Model. Without loss of generality, given an input sample $x \in \mathcal{X}$, a feature extractor f_θ embeds it into a d dimension vector $\mathbf{h} = f_\theta(x) \in \mathbb{R}^d$, where θ denotes the parameters of f. Then from $\mathbf{h}$, a classifier (including the final softmax layer) parameterized by w regresses the probability of every classes: $\mathbf{q} = g_w(\mathbf{h}) \in \mathbb{R}^C$. For simplicity, we denote the whole model by F, where $F(x) = \mathbf{q}$.
Optimization Objective. Leveraging federated learning, our objective is to obtain a global model F, which includes both the feature extractor and the classifier, to achieve the best overall prediction accuracy among all C classes. Specifically, the training is conducted on the distributed data following both the local distribution $\{d_{local}^i\}_{i=1}^N$ and the global distribution d_{global}. During the test, the final objective is to maximize $\sum_{i=1}^{C} acc^i/C$, where every class is regarded as of equal importance.

4 MOTIVATION

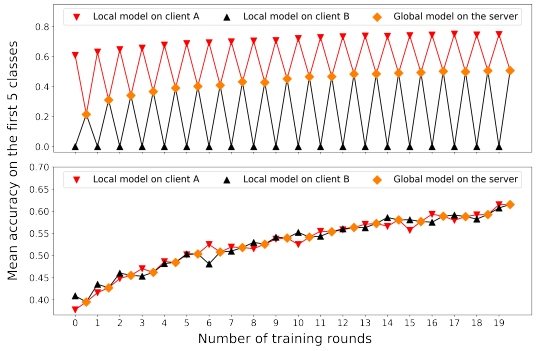
In this section, we first present a theoretical analysis which provides the theoretical foundation for our approach. In the next, we present an empirical case study that sheds key insights into the design of BalanceFL.

4.1 Theoretical Foundation

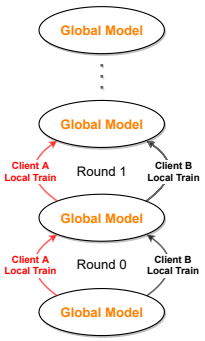
Our goal is to obtain a balanced global model under the imbalanced, long-tailed dataset in the setting of federated learning. The unique challenge caused by federated learning is the unknown global distribution, as only the model weight or the gradient is allowed to be exchanged between nodes and the server. An intuitive solution to address this challenge is to convert the task of global balancing to the node-side self-balancing. In the following, we present the theoretical analysis for this intuition.

Here, we use FedAvg [31] as the aggregation function, which has been widely used in many applications, such as next-word prediction and visual object detection. FedAvg is also shown to

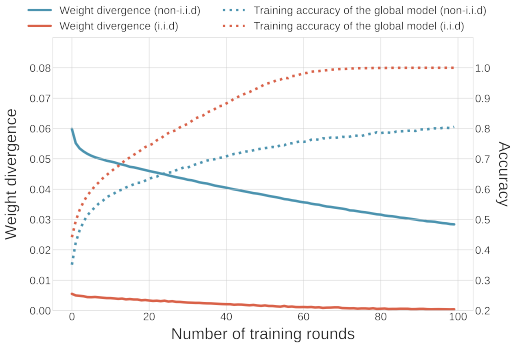
[2] For clarity, in the following of this paper, we will use superscript i to refer to the node index i, and use subscript to refer to the class index j.



(a) From top to bottom, two figures show the accuracy of models under non-iid and iid settings as federated learning goes on. Orange diamonds represent the global model after aggregation. Red and black triangles represent the model on nodes A and B just after the local update. The top figure shows a zigzag behavior, because the model on node B quickly forgets the first 5 classes (in ordinal) during the local training.



(b) An illustration of the model evolution roadmap. Different colors here correspond to those used in the legend of Figure 2(a).



(c) The solid curves represent the divergence (2-norm) of the classifier's weight of two models under two settings. The dotted curves show the training accuracy. We can see that the class missing issue enlarges the divergence, slows down the training convergence and decrease the training accuracy.

Figure 2: A motivation study to show how class missing issue can impede the knowledge accumulation in federated learning.

have several desirable theoretical properties such as the convergence guarantee under strongly convex and smooth problems [27]. Consistent with works [41, 52], there are three standard assumptions: (1) In a N client system, the global function is $F = \frac{1}{N}\sum_{i}^{N} p_i F_i$, and the function of each client F_1, F_2,..., F_N are *L-smooth*[3] and *μ-strongly convex*[4]. (2) In round t, let ξ_t^k be the data sampled from the k-th node's local data uniformly at random for $k = 1, ..., N$, where $t = 1, ..., T-1$ and T is the number of total iterations. (3) The variance of stochastic gradients and the expected squared norm of stochastic gradients in each device is bounded as:

$$\mathbb{E}\|\nabla F_k(\mathbf{w}_t^k, \xi_t^k) - F_k(\mathbf{w}_t^k)\|^2 \leq \sigma_k^2, \quad \mathbb{E}\|\nabla F_k(\mathbf{w}_t^k, \xi_t^k)\|^2 \leq G^2 \tag{1}$$

where $\mathbf{w}_t^k$ denotes the local model weights and σ_k is the variance in client k. G is the second moment bound. Then we can obtain the following equation[5], which motivates us to move the task of global balancing into the self-balancing of each client:

$$\mathbb{E}[F(\mathbf{w}_T)] - F^* \leq B \tag{2}$$

where $\mathbf{w}_T$ is the returned global model weights after T iterations. F^* is the optimal F. Specifically, Equation 2 shows the error of FedAvg has a loose upper bound B, which is negatively correlated with the number of iteration rounds T [27]. Concretely, if the data are iid, the value of $\mathbb{E}(F) - F^*$ obviously trends to zero. Although in practical scenarios, the data distribution can be uneven at a node and non-iid among nodes, we can still force the local model close to the ideal one trained under the uniformly distributed data to satisfy the assumption of Equation 2, making the learned global model close to the optimal. Therefore, our goal is transformed to how to obtain self-balanced local models, whose behavior should be similar to those trained on a uniform distributed dataset.

[3]For all $\mathbf{v}$ and $\mathbf{w}$, $F_k(\mathbf{v}) - F_k(\mathbf{w}) \leq (\mathbf{v}-\mathbf{w})^T \nabla F_k(\mathbf{w}) + \frac{L}{2}\|\mathbf{v}-\mathbf{w}\|_2^2$

[4]For all $\mathbf{v}$ and $\mathbf{w}$, $F_k(\mathbf{v}) - F_k(\mathbf{w}) \geq (\mathbf{v}-\mathbf{w})^T \nabla F_k(\mathbf{w}) + \frac{\mu}{2}\|\mathbf{v}-\mathbf{w}\|_2^2$

[5]The full version and the proof can be referred to Theorem 1 and Appendix A.3. of [27]

4.2 A Case Study

There are two challenges to obtain the aforementioned self-balanced local models. The first challenge is caused by the class missing issue defined in Section 3.2. Specifically, when a class is missing, a node has to resort to the federated learning to gain the ability to recognize this class. On the other hand, the federated learning algorithm also in turn relies on the participation of these nodes, although the data from every node alone may not play a critical role. Here, we can regard federated learning as a process of knowledge accumulation, where knowledge is extracted by local node updates and then accumulated by the server during the aggregation step. However, deep neural networks are known to be oblivious to previously learned knowledge, also termed as catastrophic forgetting [15]. For example, when training with a conventional cross-entropy loss using only the data of new classes, the model is prone to rapidly forget its knowledge on old classes. Next, we use a case study to show how the node-side class missing issue can cause catastrophic forgetting and thus impeding the knowledge accumulation of the federated learning. Specifically, we distribute the CIFAR10 dataset onto two nodes. Under the iid setting, samples of each class are equally distributed on two nodes. Under the non-iid setting, where the class missing issue occurs, the node A only has samples of the first 5 classes and node B the last 5 classes. The local epoch in this experiment is set to 5. From figure 2(a), we can see that with only few epochs of local training on the local data which misses some classes, the model will lose the recognition ability on those absent classes, as shown by the transition from orange diamonds to the black triangles. In addition, from Figure 2(c), we can observe that due to the catastrophic forgetting and local over-fitting, the model divergence between two nodes is enlarged, making the averaged model substantially biased from the optimal one.

Apart from the above knowledge forgetting problem caused by the absent classes, the differences of absolute sample numbers on those activated classes themselves lead to the imbalanced learning problem.[6] Specifically, when a node's local data only includes few samples on a class, the node itself cannot well learn the representation of this class [19], let alone share the useful knowledge that can be leveraged by other nodes through federated learning.

We note that the above two challenges have an implicit correlation in the federated learning setting. The data imbalance on activated classes impedes the knowledge extraction on local tail classes, while the class missing issue hinders the knowledge accumulation. As validated in the experiment in Table 4, leaving either of them unaddressed can hugely degrade the performance of the federated learning algorithm.

5 DESIGN OF BALANCEFL

5.1 Overview

Figure 3 shows the overall system architecture of BalanceFL. Each communication round consists of the following steps: (1) Nodes check-in with the federated server and receive the global model from the server. (2) Using the received global model, nodes perform on-device model training. (3) Once the local training is finished, nodes upload the updated model. (4) The server performs the model aggregation. Step (1), (3) are consistent with conventional federated learning approaches, while in step (2), BalanceFL features a novel node-side updating scheme that enables every node to automatically calibrate its local model as if it were trained from a uniform distribution dataset, even though the dataset itself is imbalanced in reality. In addition, our framework is agnostic to the aggregation function in step (4), thus is not restricted to the conventional aggregation method FedAvg [31].

As analyzed in Section 4.2, class missing and class imbalance of activated classes are two major impediments to the self-balancing in step (2). First, to tackle the class missing issue, we let the local model inherit the knowledge of missing classes from the downloaded global model via a distillation loss, so as to preserve the responses on those classes to prevent the catastrophic forgetting. Second, to tackle the class imbalance of activated classes, we introduce three techniques: balanced-sampling, feature-level data augmentation and smooth regularization. The balanced-sampling increases the probability of data from tail classes to be chosen to equilibrate the response of all classes during the training. The feature-level data augmentation implicitly enriches the diversity of classes with a small number of samples to avoid the over-fitting due to over-sampling. Compared with existing data-level augmentation [7, 54], this approach can be applied to arbitrary modalities. The smooth regularization penalizes the over-confident predictions for better balancing and representation learning. The above 4 steps will run iteratively before the convergence. Through node-side self-balancing, we calibrate the arbitrary local distribution to the same "pseudo-uniform" one, enabling BalanceFL to work with a low node participating ratio, which will be discussed in Section 7.4.

[6] In the following of this paper, the activated classes refer to the classes that a node has data on, and the absent classes refer to those classes that the node has no data.

5.2 Knowledge Inheritance

Although the catastrophic forgetting mentioned in Section 4.2 is a well-known drawback of deep neural networks, few works consider their side-effect in federated learning, especially when combined with the long tail learning. The solution in [39] uses the elastic weight consolidation (EWC) [23] to mitigate the non-iid issue by imposing a restriction to the changes of the weights. However, it does not consider the forgetting in a class-wise manner. In addition, previous literature shows that distillation-based methods such as learning without forgetting (LwF) [28] achieve better performance than weight regularization methods [30] such as EWC. In this paper, we follow the idea of LwF. Our insight here is that as federated learning proceeds, the server-side global model more or less has accumulated some knowledge on all classes, thus can be used as a teacher to remind the node-side local model of the knowledge on those absent classes. Therefore, during the local update, we encourage nodes to make contribution to classes they have data on (i.e., the activated classes), while discouraging the changes on predictions of absent classes to inherit the teacher's knowledge. To this end, we jointly optimize the distillation loss on absent classes as well as a regularized version of cross entropy loss (will be introduced in Section 5.3.3) on activated classes. In particular, we use $\mathbb{C}^i_{pos}$, $\mathbb{C}^i_{neg}$ to denote the set of activated classes and absent classes of node i, respectively. For each input sample x, the model will generate a C dimension prediction: $F(x) = \mathbf{q} = [q_1, ..., q_C] \in \mathbb{R}^C$, where $C = |\mathbb{C}^i_{pos}| + |\mathbb{C}^i_{neg}|$. Subsequently, the knowledge distillation loss for absent classes on node i can be given as:

$$L^i_{KI}(\hat{\mathbf{q}}', \mathbf{q}') = -\sum_{j \in \mathbb{C}^i_{neg}} \hat{q}'_j \log q'_j \tag{3}$$

where $\hat{\mathbf{q}}' = [\hat{q}'_1, ..., \hat{q}'_C]$ and $\mathbf{q}' = [q'_1, ..., q'_C]$ are modified version of the predicted probabilities (after the softmax layer) from the teacher model and the student model, respectively. Their elements are defined as:

$$\hat{q}'_j = \frac{\hat{q}_j^{1/T}}{\sum_{j=1}^{C} \hat{q}_j^{1/T}}, \quad q'_j = \frac{q_j^{1/T}}{\sum_{j=1}^{C} q_j^{1/T}} \tag{4}$$

where T is the distillation temperature. In [17], a T greater than 1 is suggested, which amplifies the originally small probabilities and soften the logits from the teacher (compared with the hard one-hot coding). Note that for preventing the catastrophic forgetting, the loss in Equation 3 only involves part of the elements of $\mathbf{q}$ (i.e., elements corresponding to absent classes) , while the loss for elements of activated classes $\mathbb{C}^i_{pos}$ will be elaborated in Section 5.3.3.

5.3 Inter-Class Balancing

In Section 5.2, the knowledge inheritance is introduced to address the class missing issue. However, for those activated classes, the data distribution is also imbalanced. In the following, we introduce three techniques to address this local inter-class imbalance.

5.3.1 Class Balanced Sampling. Without loss of generality, suppose node i has k activated classes: $\mathbb{C}^i_{pos} = \{c_1, ..., c_k\}$, the probability of

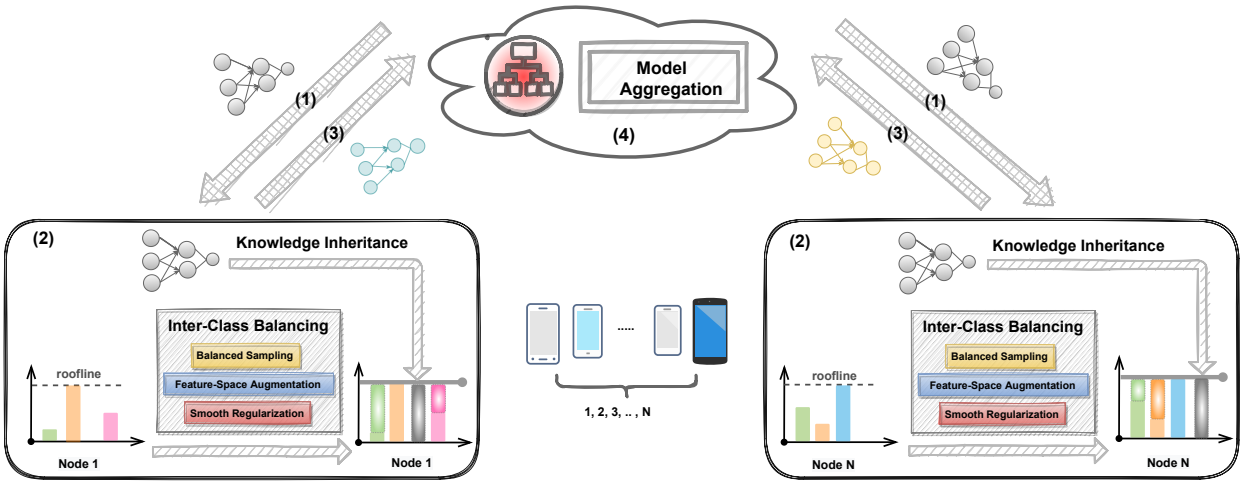


Figure 3: System architecture of BalanceFL. We achieve the self-balancing during the on-device model update in two ways. For the absent class (colored by gray), we apply the knowledge inheritance scheme to retain the knowledge from the global model. For classes that have some samples (denoted by other colors), we introduce an inter-class balancing scheme to address the imbalanced distribution issue.

sampling a data from class j can be defined as p_j :

$$p^i_{sample,j} = \frac{(m^i_j)^\gamma}{\sum_{j=c_1}^{c_k} (m^i_j)^\gamma} \tag{5}$$

where m^j_j is sample amount defined in Section 3.2, and γ is the exponential number which can determine the sampling strategy.

When γ is set to 1, Equation 5 indicates the instance-balanced sampling, which has been shown to be sub-optimal for imbalanced datasets as the gradients on few-shot classes might be overwhelmed by other classes [50], leading to unsatisfactory accuracy. In order to amplify the response of classes that have few samples, we adopt the class-balanced sampling, where γ is set to 0. In this way, we equilibrate the possibility that every class to be selected to $1/k$. One can also regard this as a two-stage sampling strategy, where we first uniformly choose a class from a set of classes $\mathbb{C}^i_{pos}$, and then pick a sample from that class uniformly.

5.3.2 Feature-Level Data Augmentation. In the above class-balanced sampling procedure, those classes with fewer samples are manually assigned with a higher probability to be selected compared to their original occurrence frequency. The side-effect of this over-sampling strategy is that some samples are repeated many times in a batch of data, resulting in the over-fitting issue. Data augmentation is known as an effective tool to enrich the diversity of data while avoiding the over-fitting. Typically, the data-level augmentation is performed by applying a wide array of domain-specific transformations to the input data, where domain expertise is required to design those transformations and to ensure that the newly synthesized data are valid. For modalities such as RGB images, there have been abundant data-level augmentation techniques [7]. However, considering that there exist a large number of sensing modalities in IoT applications and each modality requires a unique data augmentation technique, the generality and usability of the data-level augmentation is hence largely restricted. Motivated by [9], we circumvent the augmentation on input data by adopting the feature-space augmentation, which is domain-agnostic and can be applied to arbitrary modalities. Our main idea is to add a perturbation to a portion of duplicated feature vectors (due to the over-sampling) to expand the span of the feature cloud, as shown in Figure 4. Specifically, the degree of the perturbation should be first determined. We propose to transfer the overall variance of all classes to expand the feature space for those few-sample classes. We formulate the variance for node i as:

$$\Sigma^i = \frac{\sum_{j=c_1}^{c_k} m^i_j \Sigma^i_j}{\sum_{j=c_1}^{c_k} m^i_j} \tag{6}$$

where the definition of m^i_j is consistent with that in Equation 5, and Σ^i_j is the co-variance matrix of feature vectors of class j, which is with size of $\mathbb{R}^{d\times d}$, where d is the feature vector dimension defined in Section 3.2. Subsequently, a d dimensional perturbation sampled by the multivariate Gaussian distribution $N(\mathbf{0}, \Sigma^i)$ is added to original feature vector. The magnitude of Σ^i determines the degree of the perturbation.

In addition, we note that the likelihood of applying the perturbation for every class should be different. In particular, for node i, we denote the data amount of the class with most samples by $m^i_{max} = max([m^i_{c_1} ..., m^i_{c_k}])$. Through balanced sampling, where the amount of samples of every class can be regarded being aligned to m^i_{max}, every sample in class c_j in average is over-sampled by $\frac{m^i_{max} - m^i_{c_j}}{m^i_{c_j}}$ times. Those classes with fewer samples suffer a heavier repeated sampling, and therefore should be augmented with a

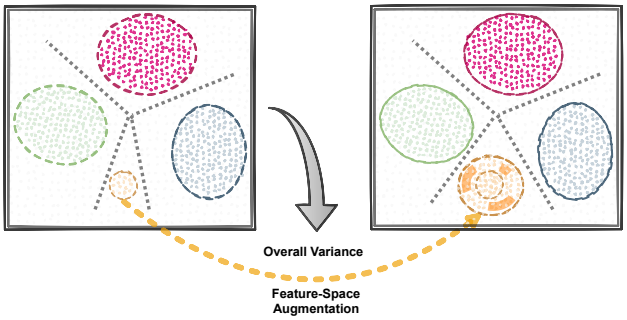


Figure 4: Illustration of the feature-space augmentation. For the class colored by brown, due to the insufficiency of the samples, the occupied feature space is too small to push other classes away, leading to the failure of a robust representation learning. We expand its occupied feature space using the overall variance obtained from all classes.

higher probability. To this end, we introduce a class-specific augmentation probability, where for class c_j, the probability of a sample to be augmented is:

$$p^i_{aug,c_j} = \frac{m^i_{max} - m^i_{c_j}}{m^i_{max}} \tag{7}$$

5.3.3 Smooth Regularization for Multi-Class Classification. In Section 5.2, for node i, the loss for absent classes $j \in \mathbb{C}^i_{neg}$ is presented, which is designed to prevent the local model from forgetting the shared global knowledge on those classes. Moreover, to endow the model the capability of discrimination on different classes, a classification loss should also be applied to activated classes $j \in \mathbb{C}^i_{pos}$. A conventional loss function for multi-class classification is the cross entropy (CE) loss, which can be defined as:

$$L^i_{CE}(\mathbf{y}, \mathbf{q}) = -\sum_{j \in \mathbb{C}^i_{pos}} y_j \log q_j \tag{8}$$

where $\mathbf{y} = [y_1, ..., y_c] \in \mathbb{R}^C$ is the one-hot ground truth label, and $\mathbf{q} \in \mathbb{R}^C$ is the predicted probability from the model, as already defined in Section 5.2. However, this CE loss may wrongly calibrate the output probabilities (a.k.a. the confidence score for every class) when the data distribution is highly biased, because the neural network is guided to be over-confident, especially on those classes that have more samples, which is often a symptom of overfitting for unwanted noises [34, 42, 57].

To alleviate this problem, apart from the above traditional CE loss, we introduce a smooth regularization term to penalize the over-confidence behavior for better generalization, which is defined as follows:

$$L^i_{Smooth}(\mathbf{q}) = \sum_{j \in \mathbb{C}^i_{pos}} q_j \log q_j \tag{9}$$

This term is also known as negative entropy. By minimizing it, we guide the neural network to output a smooth prediction, thus to promote the balance of the representations among classes. Notably, this term is specifically designed for addressing the federated imbalanced learning problem, where the key difference between ours and the standard one [34] is that ours is applied only to the activated classes $\mathbb{C}^i_{pos}$. This can prevent the regularization term from weakening the response on those absent classes, whose knowledge is still retained by the knowledge inheritance scheme proposed in Section 5.2.

5.4 Put All Things Together

We now summarize the methodology of BalanceFL from perspectives of the data and the objective function.

Training Data. Given the imbalanced training data of each node, we first apply a class-balanced sampling whose strategy is introduced in Section 5.3.1. We then perform the feature-level data augmentation to the feature vectors before the classifier with the magnitude determined by Equation 6, and with the class-wise probability given by Equation 7.

Overall Objective Function. Putting Equation 3, 8, 9 together, the overall optimization target of node i during the local training is:

$$L^i(\mathbf{y}, \mathbf{q}, \hat{\mathbf{q}}) = L^i_{KI}(\hat{\mathbf{q}}', \mathbf{q}') + L^i_{CE}(\mathbf{y}, \mathbf{q}) + \lambda_1 L^i_{Smooth}(\mathbf{q}) \tag{10}$$

where λ_1 is a balance factor. We note that the above loss function is only for one input sample for simplicity, while the one used in practice during the training should be the average over all losses of a batch of inputs.

6 DATASET

In total, our evaluation in Section 7 involves three datasets: a long-tailed version of CIFAR10 [24], a long-tailed version of Google Speech-Commands [51], and an IMU dataset for action recognition collected by ourselves. For the sake of clarity, a summary of three datasets is provided in Table 1. In the following, we will introduce in detail how each dataset is generated.[7]

6.1 CIFAR10-LT and Speech Commands-LT

6.1.1 Dataset Generation Scheme. The procedure to adapt the standard CIFAR-10 and Speech Commands datasets to long-tailed federated learning include two steps. First, we follow [3] to construct the global dataset with an overall long-tailed data distribution. We then devise a sampling mechanism to divide this global data into many sub-datasets, where each sub-dataset is used as the training data of one participant in the federated learning. We still use the original test dataset without manipulation.

Global Long-tail Construction. As mentioned in Section 3.2, we use $\Gamma = n_{max}/n_{min}$ to indicate the global imbalance degree. Given a Γ, we then let the sample amount of each class follow an exponential decay as the class index increases, i.e. $n_i = n_{max}\Gamma^{-i/(N-1)}$, where $i = \{0, 1, ..., N-1\}$.

Data Division for Nodes. We devise a sampling scheme, named τ-Sampling, to divide the global data for multiple nodes. The choice of τ will influence the degree of local class missing and the non-iidness. Specifically, we set $n_\tau = \tau n_{min}$. In every round, every node randomly samples n_τ instances without replacement from the class with the fewest data. If the data amount of a certain class is less

[7] The experiments that involve human subjects are approved by the IRB of authors' institution.

Table 1: A summary of the used datasets. The Γ is set to 100 for both CIFAR10-LT and Speech Command-LT.

Dataset	Modality	Task	# Data for Training	# Data for Test	# Classes	Imbalance Ratio
CIFAR10-LT [24]	Image	Image Recognition	12,406	10,000	10	100
Speech Commands-LT [51]	Audio	Key Word Sensing	23,463	11,005	35	100
Our Collected Dataset	IMU data	Action Recognition	6,137	3,989	8	24.8

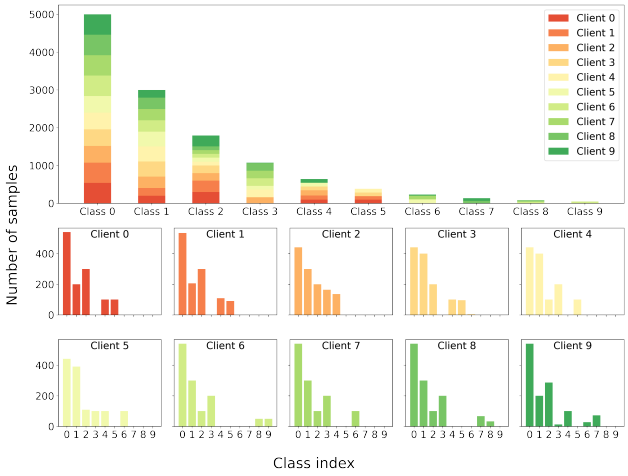


Figure 5: An illustration of our generated CIFAR10-LT dataset with 10 nodes and $\Gamma = 0.01, \tau = 2$, where the last class only includes 50 samples. The upper big plot shows the global data distribution, and the lower small plots show the local data distributed on 10 nodes.

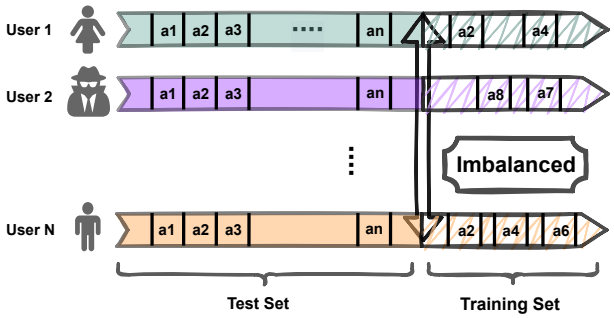


Figure 6: An illustration of our collected dataset. The letter "a" in the block means activity.

than n_τ, sample n_τ instance from this class first, and then sample the remaining instances from the class with the second smallest data amount. Rather than a generating pathological non-iid dataset, the τ-Sampling is proposed to better simulate the phenomenon that local class missing issue is more likely to happen on tail classes, as tail classes are rare in terms of both total quantity and occurrence frequency. For example, when $\tau = 1$, the class with the fewest samples will only exist on one node. A larger τ will cause a heavier class missing problem.

6.1.2 CIFAR10-LT. We use a long tailed version of CIFAR-10 [24]. The original CIFAR10 dataset includes 50,000 32×32 training images from 10 classes, with 5,000 images per class. Figure 5 shows our generated datasets of CIFAR10 with 10 nodes, where we can observe that the sample distribution of the head classes among nodes is more uniform than that of tail classes, and the class missing issue mainly occurs for tail classes, which conforms the real-life conditions.

6.1.3 Speech Commands-LT. This is a long-tailed version of the Speech Commands dataset from Google [51]. In the original dataset, there are 105,829 one-second utterances of 35 keywords collected from thousands of people. The class with the largest data amount has 3250 samples. In our experiment, we set n_{max} to 3000 and the imbalance ratio Γ to 100, so that n_{min} is equal to 30. Finally, 23,463 samples are obtained. We turn the 16 kilo samples per second (ksps) audio file into Mel Spectrogram with the FFT window size of 1024, the temporal sliding window size of 512 and the number of Mel banks of 32. By this way, every one-second audio is turn to a 32×32 one-channel image. The number of nodes is set to 10 for this dataset.

6.2 Self-Collected IMU Dataset

IMU is a sensor modality that has been widely integrated into smart phones and watches to support tasks such as sleep monitoring and action recognition. Therefore, apart from the above two synthetic long-tailed federated learning datasets, we also collect a real-life IMU dataset for action recognition. Another major difference from the two datasets above is that the tail effect of this dataset is inherent instead of introduced artificially. Specifically, the collected IMU dataset includes 8 activities: sitting, walking/pacing, lying, throwing, rummaging, stomping, hand-waving, and falling down. The data collection procedure includes two stages, as shown in Figure 6. In the first stage, volunteers are asked to only perform one specific activity in a given time period. In the second stage, we let volunteers do activities freely in the room for several minutes so that every volunteer will generate a combination of different activities at their will.

During the collection, an RGB video is also recorded whose timestamp is associated with the IMU, exclusively for annotation use. To restore the real distribution as much as possible, we do not apply any other manipulation on the training data other than removing those IMU frames that do not belong to the above 8 activities. In total, we collect 6,137 training samples from 30 people, where each person is regarded as a node in federated learning. Therefore, the number of nodes is 30. We note that the IMU dataset can also be used to simulate the cross-silo setting by grouping 30 users into multiple silos (e.g., families). Within each silo, data among users is shared. In this paper, we split the data by person, which generates more nodes and is a more challenging data allocation approach. The distributions among nodes of our collected dataset

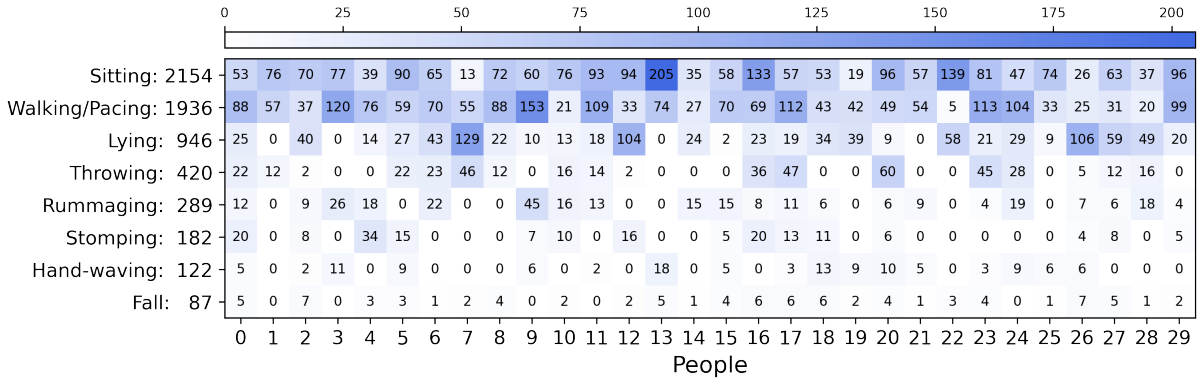


Figure 7: The data distribution of our collected IMU dataset. We can observe a long-tail phenomenon of the global data as well as the local imbalance of the local data. In addition, the global tail classes show a heavier class missing issue.

is shown in Figure 7. The sampling rate of the 9-axis IMU data is 100Hz. To capture the temporal information, we slice the IMU data sequence into multiple samples using a sliding time window of 2s. If the snippet is shorter than 200 frames, we first repeat it several times and then clip it to 200 frames. Therefore, we obtain a 1800-dimension (200 × 9) feature for each data sample.

7 EVALUATION

In this section, we conduct extensive experiments to evaluate the performance of BalanceFL. First, we evaluate the accuracy on different datasets. Then we perform the evaluation on communication overhead. In the next, we show the robustness under different participation ratios and local epochs. Last but not least, we use an ablation study to gauge the effectiveness of every proposed module. Specifically, the following six baselines are compared with BalanceFL:

(1) **FedAvg** [31]: the standard federated learning approach, where all nodes use the conventional cross entropy loss for training.
(2) **FedProx** [26]: a state-of-the-art federated learning algorithm to tackle the statistical data heterogeneity among nodes. Compared with FedAvg, an L2 regularization term is added to restrict the distance between the local model and the global model for a better convergence.
(3) **Centralized Training**: the model trained on an overall dataset aggregating all distributed data together, using the traditional cross entropy loss.
(4) **Balanced Softmax** [36]: a state-of-the-art algorithm for long-tailed recognition problem in the centralized setting, where an unbiased extension of Softmax is used to tackle the label distribution shift between the training and test dataset.
(5) **Local Training**: the model trained using only the local data at each node, with a traditional cross entropy loss.
(6) **Per-FedAvg** [55]: a recent personalized federated learning algorithm based on Model-Agnostic Meta-Learning (MAML) [13]. During the test, we let each node perform one step gradient descent on the downloaded global model for customization. As can be seen in Figures 5 and 7, the dataset shows not only class imbalance, but also data heterogeneity among clients, especially on tail classes. Compared with FedAvg, which generates a global model for all clients, Per-FedAvg is a baseline that could better addresses the data heterogeneity, although it is not specifically designed for the class imbalance problem.

The above six baselines can be divided into three groups. Baseline (1) and (2) generate only one global model for all nodes. Baseline (3) and (4) are in the centralized setting, which also generate only one model, while baseline (5) and (6) will generate one model for every node.

7.1 Implementation

We design and implement the BalanceFL prototype on a cloud server equipped with 32 virtual CPU cores and 4 Nvidia TiTAN Xp GPUs, where each provides 12 GB graphic memory. We use multiprocessing to simulate the multiple nodes in federated learning. Specifically, we let one process handle the works of the server including the node selection and model aggregation. For every node, we create one process for managing the local model updating and the data migration between the CPUs and the GPUs. The communication between the server and nodes is implemented by inter-process communication using Python3. The deep learning parts are implemented using PyTorch.

For the image recognition and key word sensing task, we use ResNet-8 [16] followed by one fully-connected layer and one softmax output layer as the deep learning model. For the task of activity recognition on the IMU data, we adopt a randomly initialized encoder-decoder-like deep neural networks (DNN) composed of three fully connected layers with hidden unit size of 256, 128, 256, respectively. In all experiments, we use Adam as the optimizer for local update with a learning rate 0.005 and a momentum 0.9. The local batch size is 64 and the default local epoch is 5. The knowledge distillation temperature in Equation 4 is set to 2, which aligns with the suggestion in [17]. For FedProx, the hyper-parameter μ to control the strength of the regularization is set to 0.05. For all federated learning methods, we limit the maximum global learning round to 200 to avoid prohibitively high communication costs.

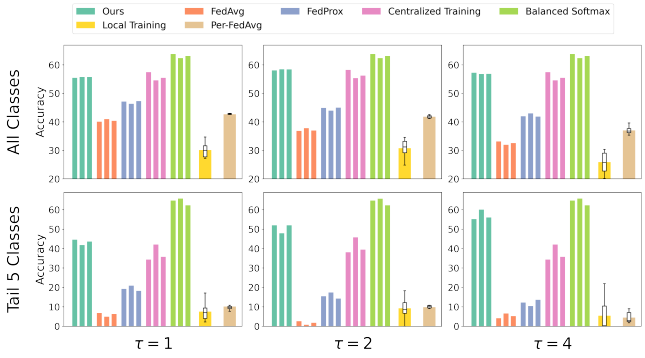


Figure 8: The accuracy of competing methods on CIFAR10-LT dataset under different τ. For local training and Per-FedAvg where one model is generated for every node, we use the Boxplot to summary the accuracy of all personalized models.

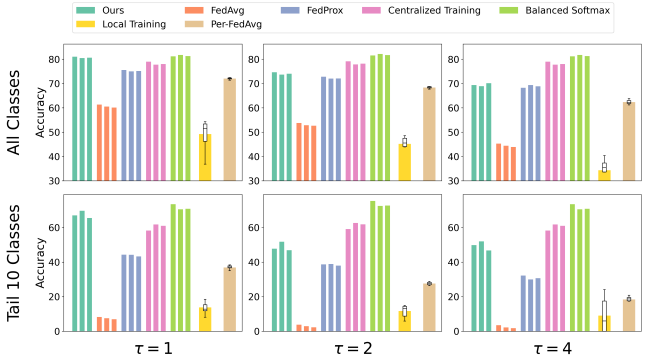


Figure 9: The accuracy of competing methods on Speech Commands-LT dataset.

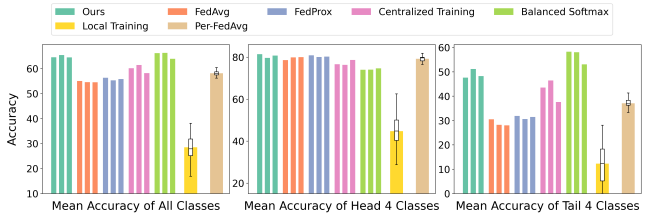


Figure 10: The accuracy of different methods on our collected IMU dataset. As data is naturally divided by person ID, there is no τ.

7.2 Accuracy on Different Datasets

Considering the random mini-batch sampling during the local training, we repeat the experiments in this Section for three times and use multiple bars to show values from different trials. The results on CIFAR10-LT are shown in Figure 8. It can be seen that our approach greatly outperforms all other federated learning approaches under all τ values. Specifically, when $\tau = 2$, our approach improves the absolute accuracy over all classes of FedAvg from 37.2% to 58.3%, where the relative improvement is up to 56.7%. Extremely, for the five tail classes, the absolute mean accuracy of FedAvg is only 1.7%, while BalanceFL achieves 50.6%.

In addition, due to the class missing issue, the local training baseline fails to gain any recognition capability on some absent classes, leading a very poor overall performance. The personalized federated learning approach (i.e., Per-FedAvg) improves the local training by taking advantage of the knowledge from other nodes. However, it still falls short in our setting, as it assumes the training data distribution and the test data distribution are identical for every node, while the goal in our setting is to train a model from an imbalanced training dataset that can achieve a high overall accuracy of all classes (i.e., the mean value of all class-wise accuracy). Notably, the centralized training achieves a much higher accuracy than the FedAvg, although neither of them considers the issue of global imbalance. There are two reasons behind. First, the knowledge of tail classes are much harder to be accumulated in the federated setting than the centralized one due to the node-side catastrophic forgetting phenomenon, as analysed in Section 4.2. Second, the data heterogeneity among nodes in federated learning (as shown Figure 5) imposes challenges to the convergence, making the finally obtained model biased from the optimal one. FedProx, to some extent, alleviates this challenge by adding a regularization term to restrict the local updates. Still, it does not consider the global imbalance, which leads to a poor performance on the tail five classes.

The results on Speech Commands-LT dataset and our collected IMU dataset are shown in Figure 9 and Figure 10, respectively. They demonstrate a consistent pattern with those on CIFAR10-LT. Regarding the accuracy over all classes, the relative improvement on FedAvg is up to 39.5%, 18.5% respectively. We note that the performance of baselines (except the local training) on the IMU dataset shows a smaller gap compared to the two synthetic datasets. This is because the tail effect is less severe. The imbalance ratio is 100 on both synthetic datasets but is 24.8 (2154/87) on the IMU dataset. We also observe that results from different trials do not show large differences on the overall accuracy of all classes. However, the accuracy of different trials shows a larger deviation on tail classes. This is because there might be a trade-off between the accuracy of head classes and tail classes, and the neural network can be biased towards either the head classes or the tail classes in different trials.

7.3 Communication and Compute Overhead

Communication costs are of great importance of the federated learning for IoT applications [31]. We measure the communication overhead by calculating the total amount of data transferred between nodes and the server before the convergence. Here, we assume that the convergence is achieved when the global model reaches 98% of the maximum accuracy at the first time. The results are shown in Table 2. We can see that BalanceFL reduces the communication cost of FedAvg by 75%, 33.6%, 68.6% on three datasets, respectively. Here, the communication costs are determined by both the size of the model and the convergence speed. On the IMU dataset where the model size is small, 30 nodes in total trigger 1.5 GB communication traffic, where the amortized one for every node is only 50 MB. We believe that such an affordable communication cost will make our framework appealing for real-world deployments of IoT systems. To intuitively visualize the improvement of our approach, we plot

Table 2: Summation of triggered communication traffic of all nodes during the whole training.

Dataset	Model	Ours	FedAvg	FedProx
CIFAR10-LT	ResNet-8	6.10 GB	24.40 GB	46.42 GB
Speech-LT	ResNet-8	17.83 GB	26.87 GB	43.64 GB
IMU	FC-3	1.50 GB	4.78 GB	5.84 GB

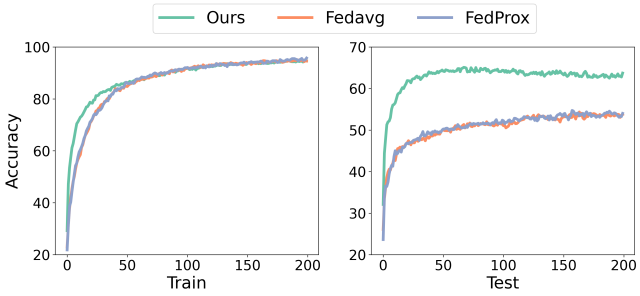


Figure 11: The curve of training and test accuracy of three approaches on our collected IMU dataset. Compared with other two baselines, BalanceFL converges to a higher test accuracy with much fewer rounds.

Table 3: Mean local update time of all clients in each federated round. The local epoch number here is set to 5. The unit is second.

Dataset	FedAvg	FedProx	Ours
CIFAR10-LT	13.6	14.5	18.2
Speech-LT	28.2	29.6	40.9
IMU	0.2	0.3	1.1

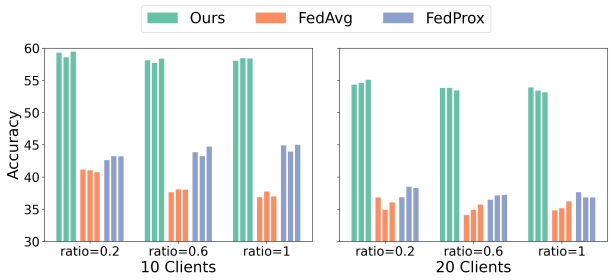


Figure 12: Accuracy of competing methods under three different participation ratios. The local epoch is 5. In the partial participation case, selected clients in each round are the same for three baselines, while they are different among trials.

curves of the training and test accuracy of the three approaches in Figure 11. It can be seen that although their training curves are very close to each other, BalanceFL quickly reaches its maximum accuracy on the test set, while FedAvg and FedProx suffer the small slope. We believe the reason is that FedAvg and FedProx lack the technique to handle the unbalancing issue. As a result, there exists a mismatch between their optimization targets (i.e., an imbalanced distribution) and the real distribution of the test data (i.e., a balanced distribution). Regarding the compute overhead, we evenly distribute all clients on 3 GPUs and record the local training time over rounds. The results are shown in Table 3. On all datasets, our approach incurs longer local update time than FedAvg. This is because the knowledge inheritance mechanism requires forward passes of the teacher model, where both the model initialization and inference cost extra time. Besides, the feature-space augmentation also consumes time to calculate the overall co-variance matrix of all features. This calculation is performed once per federated round, and is executed on the CPU in the current implementation. On the IMU dataset, although the absolute time cost is still very small, we can observe a huge relative increase of the compute time. This is due to the fast backpropagation of the lightweight IMU network on GPUs, which leads to a very small base number. In cases where the communication is the major bottleneck, the compute overhead of BalanceFL is acceptable, considering that the fewer communication rounds required compared with FedAvg.

7.4 Robustness Analysis

In this Section, we analyze the robustness of BalanceFL under different participation ratios and local epochs. Similar to Section 7.2, we repeat the experiment for three runs and use multiple bars to show results of different trials.

Different Node Participant Ratio. In real-world scenarios, it is impractical for all nodes to participate in the federated optimization in every round. For example, a mobile phone will only volunteer to participate when it is charged or under the free WiFi connection. Therefore, it is important for the framework to keep robust when some of the nodes are unavailable in some rounds. The results on CIFAR10-LT ($\tau = 2$) are shown in Figure 12. BalanceFL can keep robust under different participation ratios and consistently outperform another two generic federated learning approaches: FedAvg and FedProx. We note that, the performance under 10 nodes is better than that under 20 nodes, because each node has more data and suffer slighter local imbalance and data heterogeneity. The performance of FedAvg under participation ratio 0.2 is slightly better than that under 1 (i.e., the full participation case). This is due to the distribution gap between the training and test data, so that the distribution of aggregated data from certain clients may coincidentally be more similar to the test data distribution than the aggregated data from all nodes.

Different Local Epochs. As shown in Figure 13, we test the performance of BalanceFL under different local epochs on CIFAR10-LT ($\tau = 2$). We can observe that BalanceFL outperforms FedAvg and FedProx by a large margin under all settings. In addition, we plot the curves of the test accuracy of three approaches. As shown in Figure 14, our approach consistently achieves a higher convergence speed than other two baselines.

7.5 Ablation Study

We also perform an ablation study of BalanceFL. In total, four techniques are introduced in our framework: knowledge inheritance, balanced sampling, feature-space augmentation and smooth regularization. Table 4 shows the experimental results when some of

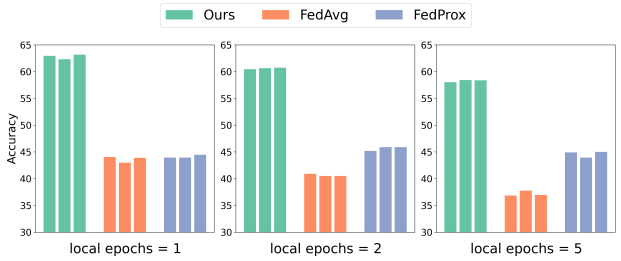


Figure 13: Accuracy under the setting of different local epochs in the full participation case.

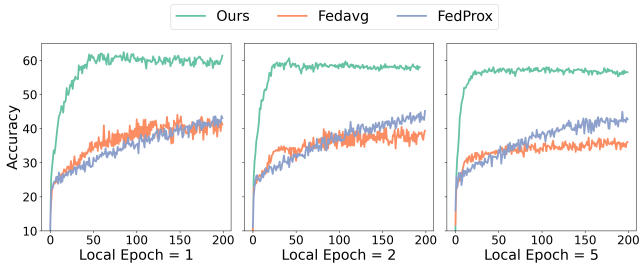


Figure 14: Curves of the test accuracy over rounds under different local epochs.

Table 4: Ablation study for BalanceFL. Without either the knowledge inheritance or the inter-class balancing technique, the performance will degrade drastically, which is consistent with the discussion in Section 4.2.

Knowledge Inheritance	Inter-Class Balancing			Accuracy
	Balanced Sampling	Feature-Space Aug.	Smooth Reg.	
✓	✓	✓	✓	58.3±0.2%
✓	✓	✓	×	57.1±0.3%
✓	✓	×	×	55.8±0.2%
✓	×	×	×	49.8±0.3%
×	✓	×	×	38.4±0.4%
×	×	×	×	37.2±0.4%

components are disabled on CIFAR10-LT dataset with $\tau = 2$. Results are reported in mean and standard deviation. Specifically, the first row represents our approach while the last row represents FedAvg. This experiment also shows that knowledge inheritance and the balanced sampling are two most important components, which tackle the class missing issue and local class imbalance, respectively.

8 DISCUSSION AND CONCLUSION

We propose BalanceFL, a long-tail federated learning framework that can robustly learn both common and rare classes from a real-world dataset, simultaneously addressing the global and local data imbalance problems. We perform the evaluation using three datasets from three different data modalities. The results show that under all datasets, BalanceFL performs significantly better than other federated approaches. Specifically, on the long-tailed version of CIFAR10, BalanceFL outperforms the FedAvg by up to 56.7% in terms of accuracy, while incurring 75% less communication overhead.

Like most current federated learning methods, our proposed framework requires the exchange of models between the server and nodes, which potentially can result in the privacy leakage. For example, by compromising the server and performing man-in-the-middle (MITM) attack, a strong attacker may infer the categories of the data on the node. Some privacy protection techniques, like differential privacy [12], thus can be combined with our framework to protect clients from attackers. In addition, some recent works [4, 53, 55] point out that, due to the latent relationship between meta-learning [13] and federated learning, the obtained global model by federated learning has high generalizability. Therefore, by applying post-processing (e.g., fine-tuning) to the global model, the local model can simultaneously achieve high generic and personalized performance. We will leave the adaptation of BalanceFL to a personalized federated learning setting as the future work. Moreover, as mentioned in Section 3.2, we assume every class is of equal importance. The extended version of it where classes have different levels of importance, will be our future work. Finally, although we address the federated long-tail learning problem from both perspectives of class missing and local class imbalance, the proposed algorithms can still be optimized. For instance, we may use adversarial training [5] for better feature-space augmentation, and we may apply extra client selection strategies [14, 25] to improve the utility and the fairness of federated learning. The large-scale validation in the cross-device federated setting will also be left for our future work.

ACKNOWLEDGMENTS

We sincerely thank the reviewers and anonymous shepherd for their valuable feedback helping us to improve this work. We thank Wenhan Lan for his help in handling the IMU dataset. This work is supported in part by the Research Grants Council (RGC) of Hong Kong under grant 14203420 and the National Natural Science Foundation of China under grant 62032021.

REFERENCES

[1] Mateusz Buda, Atsuto Maki, and Maciej A. Mazurowski. 2018. A systematic study of the class imbalance problem in convolutional neural networks. *Neural Networks* 106 (2018), 249–259. https://doi.org/10.1016/j.neunet.2018.07.011

[2] Holger Caesar, Varun Bankiti, Alex H. Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. 2019. nuScenes: A multimodal dataset for autonomous driving. *arXiv preprint arXiv:1903.11027* (2019).

[3] Kaidi Cao, Colin Wei, Adrien Gaidon, Nikos Arechiga, and Tengyu Ma. 2019. Learning Imbalanced Datasets with Label-Distribution-Aware Margin Loss. In *Advances in Neural Information Processing Systems*.

[4] Hong-You Chen and Wei-Lun Chao. 2021. On Bridging Generic and Personalized Federated Learning. *arXiv preprint arXiv:2107.00778* (2021).

[5] Tianlong Chen, Yu Cheng, Zhe Gan, Jianfeng Wang, Lijuan Wang, Zhangyang Wang, and Jingjing Liu. 2021. Adversarial Feature Augmentation and Normalization for Visual Recognition. *CoRR* abs/2103.12171 (2021). arXiv:2103.12171 https://arxiv.org/abs/2103.12171

[6] Peng Chu, Xiao Bian, Shaopeng Liu, and Haibin Ling. 2020. Feature space augmentation for long-tailed data. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIX 16*. Springer, 694–710.

[7] Ekin D. Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V. Le. 2019. AutoAugment: Learning Augmentation Strategies From Data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[8] Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. 2019. Class-balanced loss based on effective number of samples. In *Proceedings of the IEEE/CVF*

conference on computer vision and pattern recognition. 9268–9277.
[9] Terrance DeVries and Graham W Taylor. 2017. Dataset augmentation in feature space. *arXiv preprint arXiv:1702.05538* (2017).
[10] Chris Drummond, Robert C Holte, et al. 2003. C4. 5, class imbalance, and cost sensitivity: why under-sampling beats over-sampling. In *Workshop on learning from imbalanced datasets II*, Vol. 11. Citeseer, 1–8.
[11] Moming Duan, Duo Liu, Xianzhang Chen, Yujuan Tan, Jinting Ren, Lei Qiao, and Liang Liang. 2019. Astraea: Self-balancing federated learning for improving classification accuracy of mobile deep learning applications. In *2019 IEEE 37th international conference on computer design (ICCD)*. IEEE, 246–254.
[12] Cynthia Dwork and Aaron Roth. 2014. The Algorithmic Foundations of Differential Privacy. *Found. Trends Theor. Comput. Sci.* 9, 3–4 (Aug. 2014), 211–407. https://doi.org/10.1561/0400000042
[13] Chelsea Finn, Pieter Abbeel, and Sergey Levine. 2017. Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks. In *Proceedings of the 34th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 70)*, Doina Precup and Yee Whye Teh (Eds.). PMLR, 1126–1135. https://proceedings.mlr.press/v70/finn17a.html
[14] Jack Goetz, Kshitiz Malik, Duc Bui, Seungwhan Moon, Honglei Liu, and Anuj Kumar. 2019. Active federated learning. *arXiv preprint arXiv:1909.12641* (2019).
[15] Ian J Goodfellow, Mehdi Mirza, Da Xiao, Aaron Courville, and Yoshua Bengio. 2013. An empirical investigation of catastrophic forgetting in gradient-based neural networks. *arXiv preprint arXiv:1312.6211* (2013).
[16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep Residual Learning for Image Recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 770–778. https://doi.org/10.1109/CVPR.2016.90
[17] Geoffrey Hinton, Oriol Vinyals, and Jeffrey Dean. 2015. Distilling the Knowledge in a Neural Network. In *NIPS Deep Learning and Representation Learning Workshop*. http://arxiv.org/abs/1503.02531
[18] Youngkyu Hong, Seungju Han, Kwanghee Choi, Seokjun Seo, Beomsu Kim, and Buru Chang. 2021. Disentangling Label Distribution for Long-tailed Visual Recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 6626–6636.
[19] Grant Van Horn and Pietro Perona. 2017. The Devil is in the Tails: Fine-grained Classification in the Wild. *CoRR* abs/1709.01450 (2017). arXiv:1709.01450 http://arxiv.org/abs/1709.01450
[20] Yen hsiu Chou, Shenda Hong, Chenxi Sun, Derun Cai, Moxian Song, and Hongyan Li. 2021. GRP-FED: Addressing Client Imbalance in Federated Learning via Global-Regularized Personalization. *ArXiv* abs/2108.13858 (2021).
[21] Peter Kairouz, H. Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawit, Zachary Charles, Graham Cormode, Rachel Cummings, Rafael G. L. D'Oliveira, Hubert Eichner, Salim El Rouayheb, David Evans, Josh Gardner, Zachary Garrett, Adrià Gascón, Badih Ghazi, Phillip B. Gibbons, Marco Gruteser, Zaid Harchaoui, Chaoyang He, Lie He, Zhouyuan Huo, Ben Hutchinson, Justin Hsu, Martin Jaggi, Tara Javidi, Gauri Joshi, Mikhail Khodak, Jakub Konecný, Aleksandra Korolova, Farinaz Koushanfar, Sanmi Koyejo, Tancrède Lepoint, Yang Liu, Prateek Mittal, Mehryar Mohri, Richard Nock, Ayfer Özgür, Rasmus Pagh, Hang Qi, Daniel Ramage, Ramesh Raskar, Mariana Raykova, Dawn Song, Weikang Song, Sebastian U. Stich, Ziteng Sun, Ananda Theertha Suresh, Florian Tramèr, Praneeth Vepakomma, Jianyu Wang, Li Xiong, Zheng Xu, Qiang Yang, Felix X. Yu, Han Yu, and Sen Zhao. 2021. .
[22] Bingyi Kang, Saining Xie, Marcus Rohrbach, Zhicheng Yan, Albert Gordo, Jiashi Feng, and Yannis Kalantidis. 2020. Decoupling Representation and Classifier for Long-Tailed Recognition. In *International Conference on Learning Representations*. https://openreview.net/forum?id=r1gRTCVFvB
[23] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. 2017. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences* 114, 13 (2017), 3521–3526.
[24] Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. [n.d.]. CIFAR-10 (Canadian Institute for Advanced Research). ([n. d.]). http://www.cs.toronto.edu/~kriz/cifar.html
[25] Fan Lai, Xiangfeng Zhu, Harsha V. Madhyastha, and Mosharaf Chowdhury. 2021. Oort: Efficient Federated Learning via Guided Participant Selection. In *15th USENIX Symposium on Operating Systems Design and Implementation (OSDI 21)*. USENIX Association, 19–35. https://www.usenix.org/conference/osdi21/presentation/lai
[26] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. 2020. Federated Optimization in Heterogeneous Networks. In *Proceedings of Machine Learning and Systems*, I. Dhillon, D. Papailiopoulos, and V. Sze (Eds.), Vol. 2. 429–450. https://proceedings.mlsys.org/paper/2020/file/38af86134b65d0f10fe33d30dd76442e-Paper.pdf
[27] Xiang Li, Kaixuan Huang, Wenhao Yang, Shusen Wang, and Zhihua Zhang. 2020. On the Convergence of FedAvg on Non-IID Data. In *International Conference on Learning Representations*. https://openreview.net/forum?id=HJxNAnVtDS
[28] Zhizhong Li and Derek Hoiem. 2017. Learning without forgetting. *IEEE transactions on pattern analysis and machine intelligence* 40, 12 (2017), 2935–2947.
[29] Ziwei Liu, Zhongqi Miao, Xiaohang Zhan, Jiayun Wang, Boqing Gong, and Stella X. Yu. 2019. Large-Scale Long-Tailed Recognition in an Open World. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
[30] Marc Masana, Xialei Liu, Bartlomiej Twardowski, Mikel Menta, Andrew D Bagdanov, and Joost van de Weijer. 2020. Class-incremental learning: survey and performance evaluation. *arXiv preprint arXiv:2010.15277* (2020).
[31] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. 2017. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*. PMLR, 1273–1282.
[32] Aditya Krishna Menon, Sadeep Jayasumana, Ankit Singh Rawat, Himanshu Jain, Andreas Veit, and Sanjiv Kumar. 2021. Long-tail learning via logit adjustment. In *International Conference on Learning Representations*. https://openreview.net/forum?id=37nvvqkCo5
[33] Xiaomin Ouyang, Zhiyuan Xie, Jiayu Zhou, Jianwei Huang, and Guoliang Xing. 2021. ClusterFL: A Similarity-Aware Federated Learning System for Human Activity Recognition. In *Proceedings of the 19th Annual International Conference on Mobile Systems, Applications, and Services* (Virtual Event, Wisconsin) *(MobiSys '21)*. Association for Computing Machinery, New York, NY, USA, 54–66. https://doi.org/10.1145/3458864.3467681
[34] Gabriel Pereyra, George Tucker, Jan Chorowski, Łukasz Kaiser, and Geoffrey Hinton. 2017. Regularizing neural networks by penalizing confident output distributions. *arXiv preprint arXiv:1701.06548* (2017).
[35] William J Reed. 2001. The Pareto, Zipf and other power laws. *Economics Letters* 74, 1 (2001), 15–19. https://doi.org/10.1016/S0165-1765(01)00524-9
[36] Jiawei Ren, Cunjun Yu, Shunan Sheng, Xiao Ma, Haiyu Zhao, Shuai Yi, and Hongsheng Li. 2020. Balanced Meta-Softmax for Long-Tailed Visual Recognition. In *Proceedings of Neural Information Processing Systems(NeurIPS)*.
[37] Daniel Jorge Loureiro Fidalgo do Vale Rodrigues. 2019. *Risk Assessment for Alzheimer Patients, using GPS and Accelerometers with a Machine Learning Approach*. Ph.D. Dissertation.
[38] Tony Shen, Ariel Lee, Carol Shen, and C. Jimmy Lin. 2015. The long tail and rare disease research: the impact of next-generation sequencing for rare Mendelian disorders. *Genetics research* 97 (14 Sep 2015), e15–e15. https://doi.org/10.1017/S0016672315000166 26365496[pmid].
[39] Neta Shoham, Tomer Avidor, Aviv Keren, Nadav Israel, Daniel Benditkis, Liron Mor-Yosef, and Itai Zeitak. 2019. Overcoming forgetting in federated learning on non-iid data. *arXiv preprint arXiv:1910.07796* (2019).
[40] Virginia Smith, Chao-Kai Chiang, Maziar Sanjabi, and Ameet Talwalkar. 2017. Federated Multi-Task Learning. In *Proceedings of the 31st International Conference on Neural Information Processing Systems* (Long Beach, California, USA) *(NIPS'17)*. Curran Associates Inc., Red Hook, NY, USA, 4427–4437.
[41] Sebastian U. Stich. 2019. Local SGD Converges Fast and Communicates Little. In *International Conference on Learning Representations*. https://openreview.net/forum?id=S1g2JnRcFX
[42] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. 2016. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2818–2826.
[43] Canh T. Dinh, Nguyen Tran, and Josh Nguyen. 2020. Personalized Federated Learning with Moreau Envelopes. In *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin (Eds.), Vol. 33. Curran Associates, Inc., 21394–21405. https://proceedings.neurips.cc/paper/2020/file/f4f1f13c8289ac1b1ee0ff176b56fc60-Paper.pdf
[44] Alysa Ziying Tan, Han Yu, Lizhen Cui, and Qiang Yang. 2021. Towards personalized federated learning. *arXiv preprint arXiv:2103.00710* (2021).
[45] Jingru Tan, Changbao Wang, Buyu Li, Quanquan Li, Wanli Ouyang, Changqing Yin, and Junjie Yan. 2020. Equalization loss for long-tailed object recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 11662–11671.
[46] Isaac Triguero, Sara del Río, Victoria López, Jaume Bacardit, José M. Benítez, and Francisco Herrera. 2015. ROSEFW-RF: The winner algorithm for the ECBDL'14 big data competition: An extremely imbalanced big data bioinformatics problem. *Knowledge-Based Systems* 87 (2015), 69–79. https://doi.org/10.1016/j.knosys.2015.05.027 Computational Intelligence Applications for Data Science.
[47] Arijit Ukil, Soma Bandyoapdhyay, Chetanya Puri, and Arpan Pal. 2016. IoT healthcare analytics: The importance of anomaly detection. In *2016 IEEE 30th international conference on advanced information networking and applications (AINA)*. IEEE, 994–997.
[48] Lixu Wang, Shichao Xu, Xiao Wang, and Qi Zhu. 2021. Addressing class imbalance in federated learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 35. 10165–10173.
[49] Xudong Wang, Long Lian, Zhongqi Miao, Ziwei Liu, and Stella Yu. 2021. Long-tailed Recognition by Routing Diverse Distribution-Aware Experts. In *International Conference on Learning Representations*. https://openreview.net/forum?id=D9I3drBz4UC
[50] Yu-Xiong Wang, Deva Ramanan, and Martial Hebert. 2017. Learning to Model the Tail. In *Proceedings of the 31st International Conference on Neural Information*

Processing Systems (Long Beach, California, USA) *(NIPS'17)*. Curran Associates Inc., Red Hook, NY, USA, 7032–7042.

[51] Pete Warden. 2018. Speech commands: A dataset for limited-vocabulary speech recognition. *arXiv preprint arXiv:1804.03209* (2018).

[52] Hao Yu, Sen Yang, and Shenghuo Zhu. 2019. Parallel restarted sgd with faster convergence and less communication: Demystifying why model averaging works for deep learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 5693–5700.

[53] Tao Yu, Eugene Bagdasaryan, and Vitaly Shmatikov. 2020. Salvaging Federated Learning by Local Adaptation. arXiv:2002.04758 [cs.LG]

[54] Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. 2018. mixup: Beyond Empirical Risk Minimization. In *International Conference on Learning Representations*. https://openreview.net/forum?id=r1Ddp1-Rb

[55] Michael Zhang, Karan Sapra, Sanja Fidler, Serena Yeung, and Jose M. Alvarez. 2021. Personalized Federated Learning with First Order Model Optimization. In *International Conference on Learning Representations*. https://openreview.net/forum?id=ehJqJQk9cw

[56] Songyang Zhang, Zeming Li, Shipeng Yan, Xuming He, and Jian Sun. 2021. Distribution Alignment: A Unified Framework for Long-Tail Visual Recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2361–2370.

[57] Zhisheng Zhong, Jiequan Cui, Shu Liu, and Jiaya Jia. 2021. Improving Calibration for Long-Tailed Recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 16489–16498.