# FedGCS: Addressing Class Imbalance in Long-Tail Federated Learning

Guozheng Liu[1,2], Wei Zhang[1,2(✉)], Huiling Shi[1,2], Lizhuang Tan[1,2], Chang Tang[1,2], and Meihong Yang[1,2(✉)]

[1] Key Laboratory of Computing Power Network and Information Security, Ministry of Education, Shandong Computer Science Center (National Supercomputer Center in Jinan), Qilu University of Technology (Shandong Academy of Sciences), Jinan, China
10431210755@stu.qlu.edu.cn, {wzhang,shihl,tanlzh,yangmh}@sdas.org
[2] Shandong Provincial Key Laboratory of Computer Networks, Shandong Fundamental Research Center for Computer Science, Jinan, China
tangchang@cug.edu.cn

**Abstract.** Federated learning is a privacy-preserving distributed machine learning method, which facilitates clients to cooperate in training a shared model while safeguarding original data. The different distribution and quantity of training data between clients can pose significant challenges, such as data heterogeneity and class imbalance, which can greatly influence the performance of the shared model. Although many methods have been proposed to eliminate the deleterious influence of non-IID data, existing solutions usually do not perform well on tail-classes owing to the absence of attention for the long-tail distribution. We present a long-tail federated learning framework FedGCS, which can solve the global and local class imbalance problem via generic to compensate for specific. Specifically, clients separate features from the training data based on the class activation map and selectively fuse the separated class-specific features and class-generic features to restore the distribution of tail-classes. We also design a loss function—TailDistillation Loss to lessen the bias of the classifier towards head-classes. To appraise the effectiveness of FedGCS, we adapted multiple benchmark datasets to the long-tail federated learning setting. Experiments indicate that the FedGCS is an useful method, and is superior to previous approaches.

**Keywords:** Federated learning · Long-tail learning · non-IID

## 1 Introduction

Federated learning (FL) makes clients train a shared model together, abstaining from uploading private data to server. It can settle the issue of data security and privacy protection, while reducing resource consumption on the central server. FL has a good application prospect in numerous areas, such as medical care [30],


A. Zaslavsky et al. (Eds.): MobiQuitous 2023, LNICST 593, pp. 223–238, 2024.
https://doi.org/10.1007/978-3-031-63989-0_11

financial services [38], Internet of Things [23] and network resource scheduling [6,7].

In standard FL, for each iteration, server will arbitrarily choose a subgroup of all participants and distribute current model. These chosen clients will then train the received model using a private dataset and send their gradient adjustments to server for aggregation. Finally, the server will return updated shared model to all participants. But during FL training, a severe reality problem is heterogeneity arising from non-IID data between varied clients [18]. In the FL situation, the data owned by various clients have different sources and may contain their preferences, which leads to distinct distributions and quantities between clients.

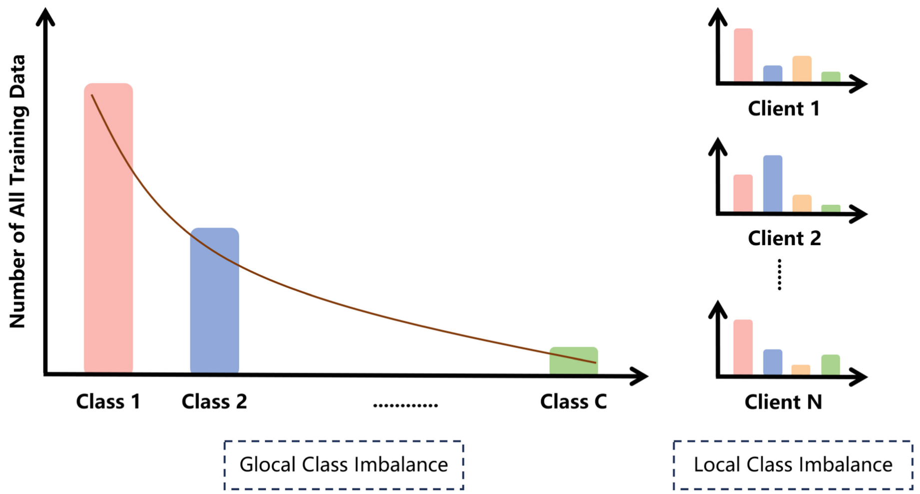


**Fig. 1.** Global class imbalance and local class imbalance in FL. Research purpose is to alleviate the imbalance problem in training data below the long-tail federated learning setting.

Additionally, actual world data usually exhibit long-tail distribution characterized by severe class imbalance, i.e., the samples for certain classes is much larger than others, for example, the number of patients with different diseases in medical diagnoses is significantly different, and the number of fraudulent transactions in credit card transactions is much lower than normal transactions. In the FL scenarios, class imbalance may exist globally and locally [31], as illustrated in Fig. 1. Among them, global class imbalance means that the amalgamation of clients local data follows a long-tail distribution, and local class imbalance denotes that each class samples on every client is extremely uneven. One radical scenario of local class imbalance is class missing, which means that some classes do not exist, and this problem is likely to occur in those global minority classes. There may also be situations where certain classes are head-classes on some clients and tail-classes on other clients, resulting in data heterogeneity between clients. When class imbalance exist, the local model may be biased in the process of learning and generalization, which is further inclined towards learning head-classes and neglects tail-classes importance, bringing about the dramatic diminishment in the recognition and classification accuracy of tail-classes in the model. However, in many practical cases, tail-classes may play an important role that far exceeds their proportion in the sample, for example, health monitoring equipment needs to be more sensitive to abnormal conditions than normal [20], and the auto drive system is more important to accurately predict the status of traffic lights than to predict the names of surrounding buildings [12].

Recently, lots of approaches have been presented to address the long-tail issue. One common method is class rebalancing, which aims to decrease the effect of imbalance through resampling or reweighting technique. In addition, although the overall performance can be improved, this type of method usually sacrifices head-classes performance for tail-classes performance, and cannot fundamentally settle the state of lacking data information in the long-tail problem, so this may not be desirable in practical applications. Moreover, most of these methods are based on the premise that the global data distribution is observable, which does not apply to FL where data cannot be exported locally. At present, there are less studies on long-tail federated learning, and most of them do not notice the unified treatment of global class imbalance and local class imbalance.

Within this research, we present a long-tail federated learning framework FedGCS, which can solve the class imbalance problem via generic to compensate for specific. Specifically, first, the confusing class relative to the tail class is obtained locally on each client through similarity between classes. Then, based on the class activation map (CAM) [28,39], the features of the training data are partitioned into class-generic features with little difference between classes and class-specific features with significant difference between classes. Finally, over the FL training phase, the class-specific features of tail-class are combined with the class-generic features of the corresponding confusing classes to obtain novel data features of tail-classes. Using this method, the local model performance on the client can be improved, thereby improving the final aggregated global model performance. Furthermore, when the data present a long-tail distribution, it will cause the classifier to lean more towards identifying head-classes, thus ignoring tail-classes. We introduce TailDistillation Loss, in order that the client local model can further retain the knowledge about tail-classes in shared model over the training phase, decreasing the classifier's inclination for head-classes. We evaluated the proposed FedGCS on CIFAR-10/100-LT [16]. A considerable amount of experiments indicate that FedGCS outperforms the existing FL approaches in image classification tasks with the long-tail distribution. Subsequently, we provide an overview of the contributions made in this study.

- We present a long-tail FL framework FedGCS, which does not need to know the information of global data distribution, enhances tail-classes features in client, and we introduce TailDistillation Loss to correct the model during training phase.
- We perform comprehensive experiments on two long-tail datasets. Our method is proved to be better than other FL Approaches in long-tail distribution image classification tasks.
- Our work demonstrates the importance of recognizing the long-tail distribution as early as possible and taking timely measures during FL.

## 2 Related Work

### 2.1 Federated Learning

FL was presented in [24], including FedAvg, the most basic and classic algorithm, which obtains the aggregation model by iterating and averaging parameters from the distributed client model, thus avoiding the need to access the client local data. For the data heterogeneity in FL, there has been a lot of work. On the one side, local training is stabilized by fine-tuning the disparity between models for client and server [15,19,32]. On the other side, it aims to optimize the effectiveness of model aggregation [2,34,35]. Among them, knowledge distillation [9] has become an effective solution [11,17,21], enriching the shared model through integrating local model information and alleviating model drift problem caused by heterogeneity. Although the above methods have solved to some extent the problem of data heterogeneity, the final model obtained exhibits unsatisfactory performance on tail-classes, attributed to overlooking the global distribution.

At present, only few studies have recognized the issue of decreased model accuracy resulting from class imbalance (not specific to long-tail distribution). Duan et al. [4] proposed the Astraea framework, which mitigates class imbalance through information augmentation according to global information distribution and multi-client rescheduling on basis of intermediaries. Shang et al. [29] presented an approach of applying federated features for classifier retraining to solve the union issue of data heterogeneity and long-tail distribution. Wang et al. [36] introduced a monitoring solution capable of deducing the constitution of training samples in every FL iteration, aimed at alleviating the influence of global class imbalance. In our scenario, local data presents a long-tail distribution, where some important classes having only small or even no samples. Therefore, it is necessary to design a new algorithm specifically for this unbalanced property to address this problem.

### 2.2 Long-Tail Learning

In the actual world, training samples always exhibit severe class imbalance, known as long-tail distribution, which has attracted widespread attention in the field of deep learning [37]. The objective of long-tail learning is to develop high-performance models on the data with long-tail distribution. Currently, some solutions have been proposed, including class rebalancing, information augmentation, and module improvement. The methods based on class rebalancing aim to alleviate the impact of class imbalance through resampling or reweighting technique. Feng et al. [5] suggested monitoring model training across various classes by averaging classification prediction scores. Ren et al. [25] presented to utilize the training label oftenness to reweight prediction logits, in order that the label can mitigate the bias of class imbalance before calculating final losses. The methods based on information augmentation are designed to enhance the model performance of long-tail learning though introducing additional information. Chu et al. [3] proposed to enhance tail-classes with few samples by learning features from head-classes with ample samples. Liu et al. [22] proposed using the geometric information of classifiers with relatively large head-classes to enhance classifier weight assigned to tail-classes, thereby improving tail-classes performance. The methods based on module improvement aim to tackle long-tail problems though ameliorating network modules. Kang et al. [13] introduced an innovative decoupled training approach, to gain a more stable feature space, resulting in improved performance in long-tail learning. Zhou et al. [40] proposed a novel strategy to address long-tail recognition. In detail, traditional studying branch employs evenly extraction to mimic a long-tail distribution, whereas re-balancing branch employs a reverse extractor to augment the sampling of tail-classes. But most of long-tail learning methods need information on global class distribution, which is difficult to apply directly to FL.

# 3 Proposed Method

## 3.1 Problem Formulation

**Problem Setting.** We discuss a standard FL setup where $K$ clients each hold a potential non-IID dataset $\mathcal{D}^1, \mathcal{D}^2, \ldots, \mathcal{D}^K$, and the union of all these datasets $\mathcal{D} = \cup_k \mathcal{D}^k$ follows a long-tail distribution, i.e. $D \in (\mathcal{X}, \mathcal{Y})$ and $\mathcal{Y} \in \{1, \ldots, C\}$. Setting $n_c^k$ be the quantity of samples for class $c$ on client $k$, and $n_c = \sum_{k=1}^{K} n_c^k$. Wherein all classes are ordered in a diminishing manner based on respective sample numbers, that is, if $c_1 < c_2$, there will be $n_{c_1} \geq n_{c_2}$. We introduce $r \in (0, 1)$, representing the proportion of samples from head-classes to overall sample numbers. Additionally, we specify the global imbalance factor as the quotient of the sample count in the largest class divided by the sample count in the smallest class, i.e., IF$= \frac{n_1}{n_C}$. We also use the Dirichlet distribution factor $\alpha$ to regulate heterogeneity extent. The model we use in FL is a neural network $\phi_{\mathbf{w}}$ with parameter $\mathbf{w} = \{\mathbf{u}, \mathbf{v}\}$, which comprises a feature extractor $f_{\mathbf{u}}$ and a classifier $h_{\mathbf{v}}$. The parameter representation of the local model for client $k$ is $\mathbf{w}_k$.

**The FedAvg Baseline Algorithm.** FedAvg [24] is the first and most classic FL algorithm. In the $t$-th round of communication, Initially server arbitrarily chooses a subgroup $S^t$ from all clients and sends a shared model $\mathbf{w}^t$ to them. Then the client uses local data $\mathcal{D}^k, k = 1, \ldots, K$ to perform iterative optimization with learning rate of $\eta$ to further adjust the model:

$$\mathbf{w}_k^{t+1} \leftarrow \mathbf{w}_k^t - \eta \nabla_{\mathbf{w}} \ell \left( \mathbf{w}^t ; \mathcal{D}^k \right). \tag{1}$$

After local updating, clients uploads their updated models to server. At last, server aggregates models through weighted averaging to obtain shared model for the $t + 1$ communication round:

$$\mathbf{w}^{t+1} = \sum_{k \in \mathcal{S}^t} \frac{\left| \mathcal{D}^k \right|}{\sum_{k \in S^t} \left| \mathcal{D}^k \right|} \mathbf{w}_k^{t+1}. \tag{2}$$

### 3.2 Framework of FedGCS

Based on the above assumptions, we propose FedGCS to solve the global and local long-tail distribution problem in FL. It is a two-stage training scheme based on FedAvg. In the stage one, clients jointly train a shared model according to standard FL training steps. In the stage two, each client performs feature separation on the training data locally based on the CAM and selectively fuses the separated class-specific features and class-generic features to generate augmented samples of tail-classes. On this basis, the local classifier is fine-tuned and uploaded to server to further update shared model. Our goal is to enrich data multiplicity and prevent overfitting through feature-level information augmentation. In addition, in this scheme, we also introduce TailDistillation Loss, which allows the client local model to further preserve the information about tail-classes in the shared model, to reduce classifier bias towards head-classes. The framework of FedGCS is depicted in Fig. 2.

Specifically, in the Phase-I, we use images from local dataset of the client to train the model issued by server, and upload updated gradient to server for aggregation. After multiple iterations, we obtain a global model. To subsequently compute the CAM, we select a network structure (ResNet [8]) with a single fully connected layer. Numerous deep convolutional neural networks belong to this category, such as DenseNet [10], MobileNet [26] , and EfficientNet [33].

In the Phase-II, the problem we need to solve is, how to select classes for tail-classes that can extract generic features from them. We believe that classes that are relatively near to specified tail class in the distribution, meaning they are more likely to be confused with it, which has a significant impact on restoring the distribution of tail-classes. Specifically, every client use the shared model obtained in the Phase-I to predict classification scores for all classes, excluding the specified tail class, for each training sample, and by sorting the average classification scores, find its confusing classes $C_f$.

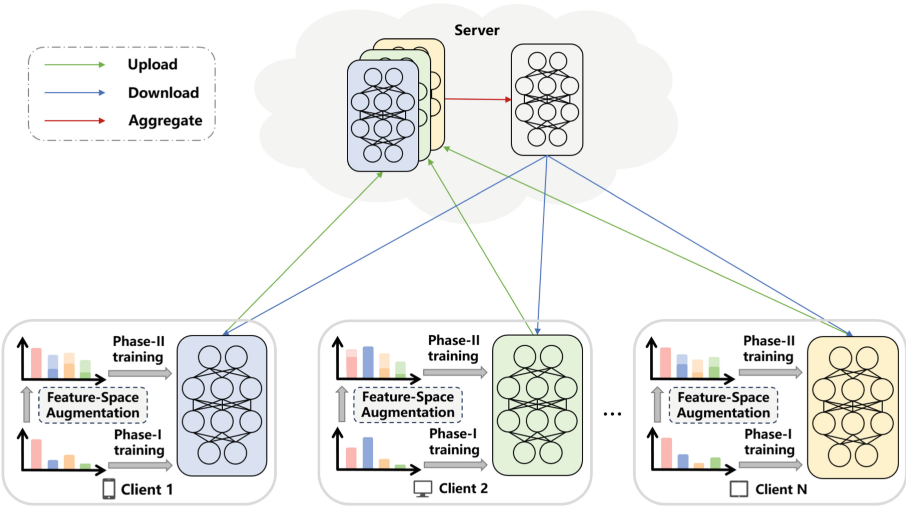


**Fig. 2.** The framework of FedGCS.

We also need to segment the features of the training data in each client, distinguishing between class-specific and class-generic features that can be utilized to restore the tail class distribution. Motivated by current work on attention and visual interpretation [28,39], we first calculate the score $y_c$ of class $c$ regarding feature activation map $A^l$ gradient of the convolution layer, i.e. $\frac{\partial y^c}{\partial A^l}$. Subsequently, we carry out global avgpooling on backflow gradients, bringing about neuron importance weights $\alpha_l^c$:

$$\alpha_l^c = \frac{1}{Z} \sum_i \sum_j \frac{\partial y^c}{\partial A_{ij}^l}. \tag{3}$$

Then, we weight the forward activation maps, and subsequently apply a ReLU to get the CAM $M_c$ of class $c$:

$$M_c = \mathrm{ReLU}\left( \sum_l \alpha_l^c A^l \right). \tag{4}$$

Finally, specifying a threshold of $0 < \xi < 1$. We can disintegrate $M_c$ into $M_c^s$ and $M_c^g$, and get class-specific and class-generic features, this formula is shown below:

$$M_c^s = \mathrm{sgn}(M_c - \xi) \odot M_c, \tag{5}$$

$$M_c^g = \mathrm{sgn}(\xi - M_c) \odot M_c, \tag{6}$$

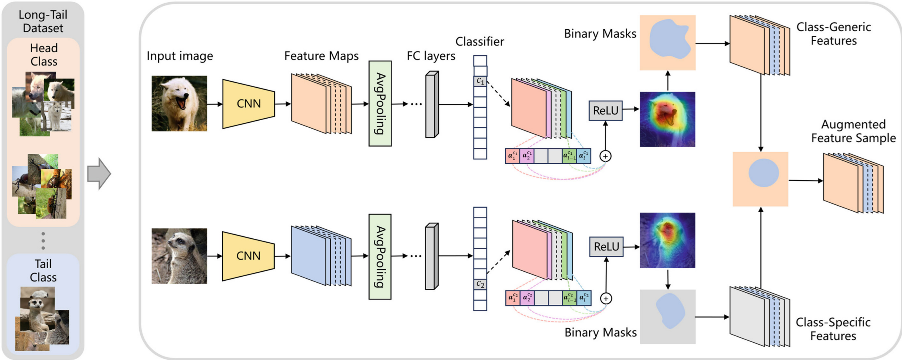


**Fig. 3.** The procedure of feature enhancement on client. Fusing the class-specific features of tail-classes images with the class-generic features of the corresponding confusing classes images in feature space to produce augmented samples of tail-class.

Next, we use the class-specific features separated from tail-class images and the class-generic features separated from the corresponding confusing class images for feature fusion. To mitigate noise and potential bias, we employ a linear fusion approach in the feature space, guiding the fusion process by randomly generating combination ratios to produce enhanced samples for tail-class, restoring its inherent data distribution, as shown in Fig. 3. For per sample in tail-class, altogether produce $N_a$ augmented samples. Finally, the generated augmented samples are used locally on each client to fine-tune the classifier and uploaded to server to further update shared model.

In addition, we also propose to decrease classifier bias toward head-classes by introducing TailDistillation Loss (denoted by $L_{TD}$). Specifically, we consider using the CrossEntropy Loss function, which is widely used in multi-class classifiers, as the basic term to define $L_{TD}$ as:

$$\begin{aligned} L_{TD} &= (1-\lambda) * \mathrm{CE}\left(gt, \sigma\left(z_s; T=1\right)\right) \\ &+\lambda * \mathrm{CE}\left(\sigma\left(z_t^{\prime}; T=\tau\right), \sigma\left(z_s^{\prime}; T=\tau\right)\right), \end{aligned} \tag{7}$$

where $gt$ is the ground truth label (one-hot). $z_s$ represents local model's predicted result on local data, $z_t^{\prime}$ represents the global model's predicted result on local tail-classes data, and $z_s^{\prime}$ represents the local model's predicted result on local tail-classes data. $\sigma$ is the softmax function parameterized by the temperature $T$, and $\lambda$ is a balancing hyperparameter, which ranges from $[0,1]$.

# 4 Experiment

## 4.1 Experiment Settings

We choose two baseline datasets: CIFAR-10 and CIFAR-100 [16], and according to [1], shape them into long-tail versions (CIFAR-10/100-LT) with varied IFs (10, 50, and 100), to fit into long-tail federated learning setting. Moreover, utilizing Dirichlet distribution to partition the non-IID data between different clients and set the value of $\alpha$ to 0.5. We use ResNet-8, which containing a single fully connected layer. To guarantee an equitable comparison, we have achieved all FL methods for comparison applying same model. By default, we run a total of 200 rounds of global communication with a total of 20 clients, possessing an active user percentage of $C = 50\%$ per round. For per sample in client tail-class, we choose $N_a = 3$. We set the local training batch size to 128, learning rate to 0.01, and the optimizer to SGD.

**Table 1.** Test accuracy (%) for different FL methods and FedGCS on CIFAR-10/100-LT with diverse IFs.

| Family | Method | CIFAR-10-LT | | | CIFAR-100-LT | | |
|---|---|---|---|---|---|---|---|
| | | IF = 10 | IF = 50 | IF = 100 | IF = 10 | IF = 50 | IF = 100 |
| Heterogeneity-oriented FL methods | FedAvg | 59.53 | 54.95 | 53.32 | 38.62 | 28.47 | 25.16 |
| | FedProx | 60.22 | 56.43 | 53.45 | 38.93 | 28.52 | 25.64 |
| | FedNova | 61.45 | 56.79 | 55.62 | 39.31 | 28.19 | 26.22 |
| Imbalance-oriented FL methods | Fed-Focal Loss | 59.39 | 53.67 | 50.32 | 35.17 | 26.74 | 24.58 |
| | Ratio Loss | 60.96 | 58.26 | 55.93 | 39.45 | 29.68 | 27.03 |
| | FedAvg+$\tau$–norm | 51.12 | 48.81 | 46.67 | 31.02 | 23.09 | 20.32 |
| Proposed Method | FedGCS | **64.57** | **62.32** | **61.75** | **40.83** | **33.54** | **31.76** |

## 4.2 Comparison with Other Methods

To validate the efficaciousness of FedGCS, we compare the proposed method with these heterogeneity-oriented FL methods: FedAvg [24], FedProx [19] and FedNova [35]. In addition, we compare imbalance-oriented FL methods: Fed-Focal Loss [27], Ratio Loss [36], and FedAvg with $\tau$-norm [14].

We evaluate FedGCS on CIFAR-10/100-LT, as depicted in Table 1. FedGCS achieved the optimal test accuracy in all scenarios. Compared to the baseline FedAvg, when IF = 100, FedGCS has the highest performance gain, approximately 8.4% for CIFAR-10-LT and 6.6% for CIFAR-100-LT. It implies that FedGCS has excellent generalization ability. For heterogeneous-oriented FL methods, such as FedProx, exhibit performance similar to FedAvg, as they primarily address non-IID data while disregarding the long-tail distribution. For class imbalance-oriented FL methods, such as Fed-Focal Loss, perform well compared to FedAvg in certain situations. However, when compared to FedGCS, there remains a performance gap as these methods only address global class imbalance concerns and overlook data heterogeneity issues.

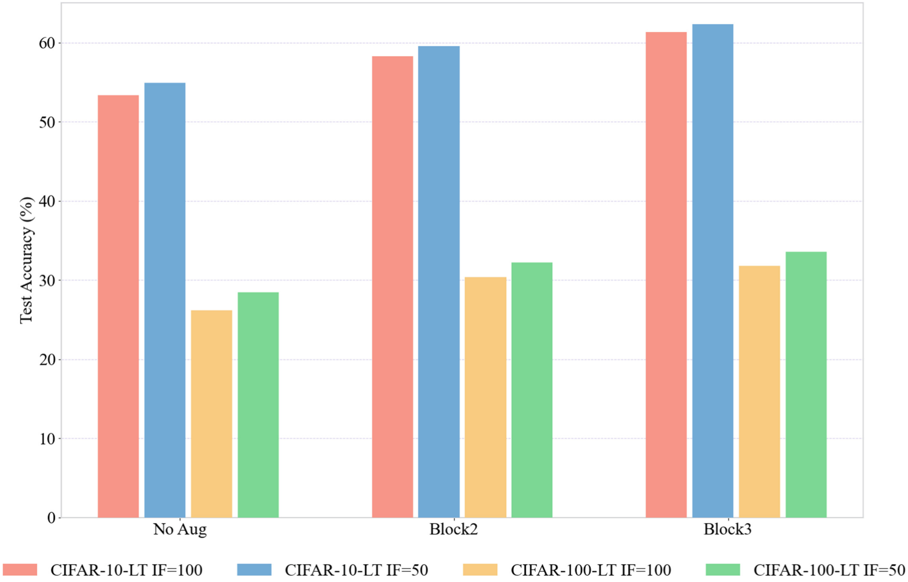


**Fig. 4.** Test accuracy though applying feature augmentation at diverse depths of ResNet-8.

### 4.3 Model Validation

**Ablation Analysis on Applying Feature Augmentation at Diverse Depths of Network.** The ResNet-8 model used for the CIFAR-10/100-LT dataset composes of three convolutional blocks. We perform feature enhancement in the model, subsequent to the final two convolutional blocks. After enhancing the features after Block2 and Block3 and generating augmented samples, the test accuracy after fine-tuning in the Phase-II is shown in Fig. 4. It is obvious that among different datasets and imbalanced factors, feature enhancement after Block3 has the best effect. Feature maps nearer to the input end encompass greater spatial details, but they may introduce extra artifacts. To address this issue, we pass the features generated by Block3 directly to the global pooling layer, removing noise caused by augmentation. Therefore, we choose to apply feature enhancement after Block3. We evaluate the performance without the application of enhancement, referred to as "No Aug" in the figure.

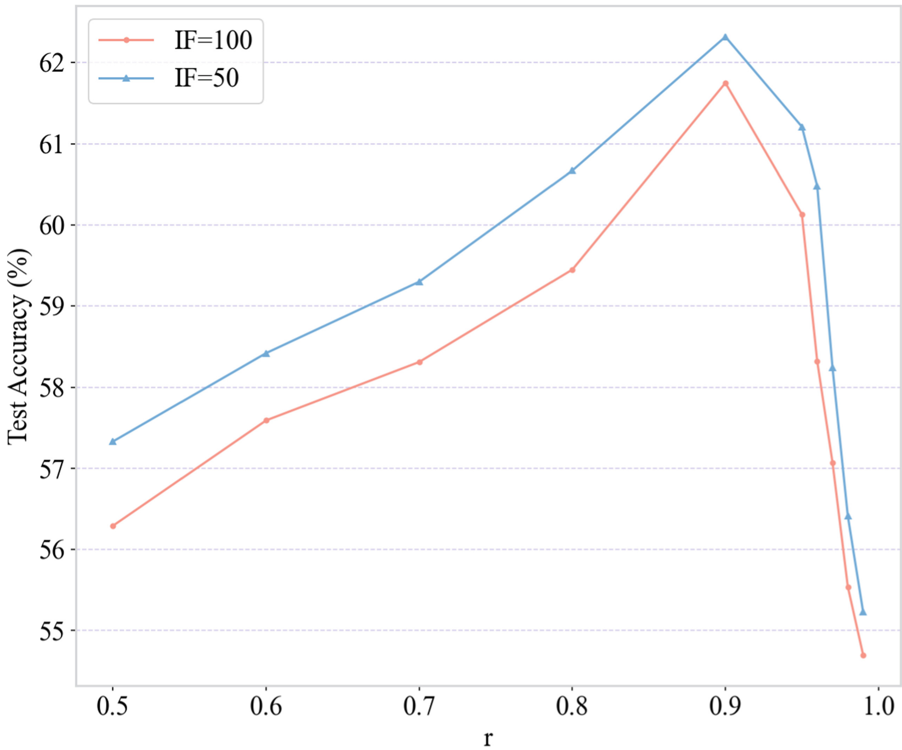


**Fig. 5.** FedGCS performance across varied $r$ values on CIFAR-10-LT.

**Analysis of Hyperparameter Sensitivity.** We assess the consequence of choosing the varied values of $r$ on the final classification accuracy in Fig. 5. The curves exhibit peaks near $r = 0.9$. On the ascending side of the peak, only less classes are utilized as head-classes, making it insufficient to capture class-generic features. On the descending side of the peak, only less classes are employed as tail-classes, leaving classes with insufficient samples unable to use augmented samples for fine-tuning. Moreover, we select the smallest $r$ value that meets the condition in every experiment. We also study the impact of the equilibrium factor $\lambda$ in the loss function. This hyperparameter controls the distillation intensity of the tail class in Eq. 7. In Fig. 6, it is apparent that, the greater the $\lambda$ value, the intensified the model accuracy degradation. This shows that distillation of tail-classes data alone is not sufficient to obtain an excellent model.

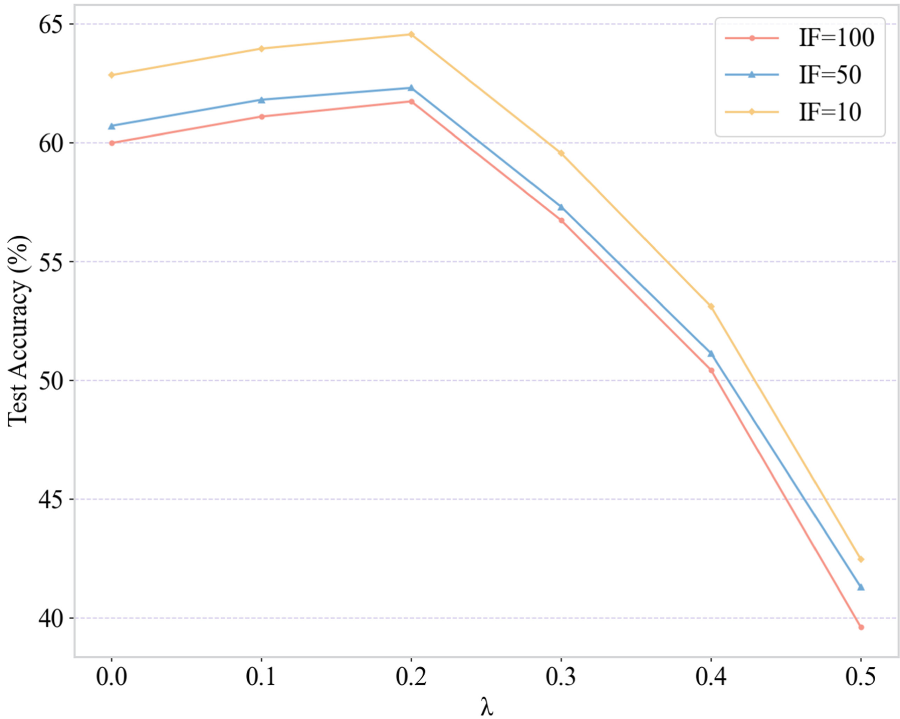


**Fig. 6.** FedGCS performance across varied $\lambda$ values on CIFAR-10-LT.

**Impact of the Extent of Non-IIDness.** Fig. 7 additionally illustrates the test accuracy of the three methods across varied extents of non-IIDness. It is observable that the performance of every method declines with increasing non-IIDness extent. When the $\alpha$ value decreases from 1 to 0.01, the compared methods exhibit a greater decrease in performance compared to FedGCS.

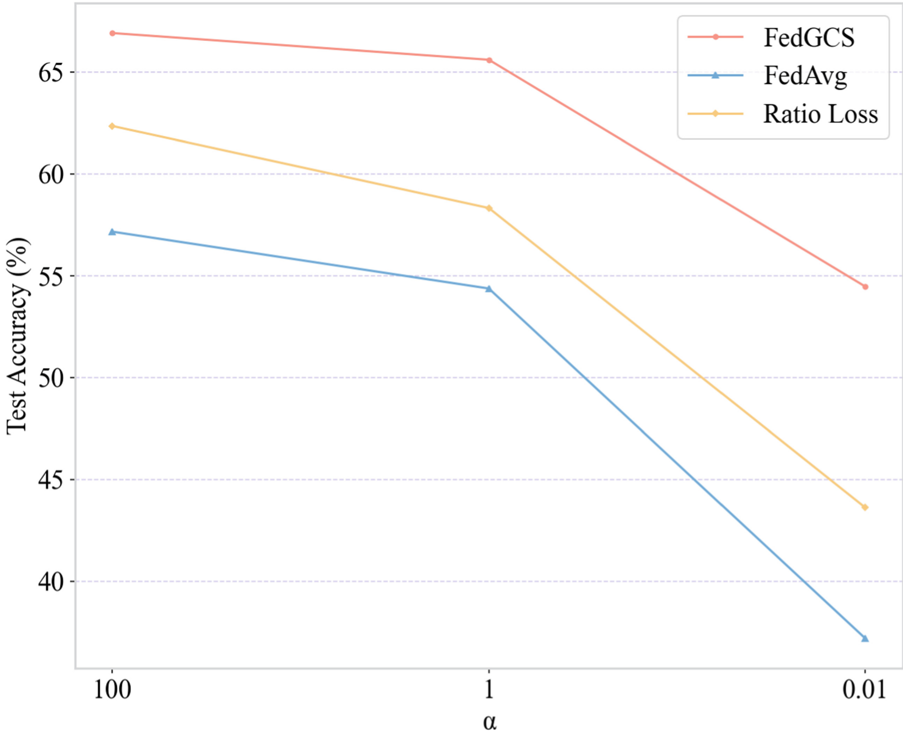


**Fig. 7.** FedGCS performance across varied $\alpha$ values on CIFAR-10-LT with IF=100.

## 5 Conclusion

In this study, we present FedGCS to deal with the global and local class imbalance problem in long-tail FL. On the client, the corresponding confusing classes for local tail-classes are computed on basis of global model. Then, feature separation is performed on the local data using the class activation map, and the class-specific features of the tail-classes are integrated with the class-generic features of the corresponding confusing classes to restore tail-classes distribution. Furthermore, we construct TailDistillation Loss to mitigate the effects of global and local class imbalance. Numerous experiments demonstrate that FedGCS outperforms the current heterogeneity-oriented and imbalance-oriented FL approaches significantly when dealing with a long-tail distribution.

**Acknowledgements.** This work was supported in part by the Shandong Provincial Natural Science Foundation under Grant No. ZR2020LZH010 and No. ZR2022 LZH015, the Project of Key R&D Program of Shandong Province under Grant No. 2022CXGC20106, the Pilot International Cooperation Project for Integrated Innovation of Science, Education and Industry of Qilu University of Technology (Shandong Academy of Sciences) under Grant No. 2022JBZ01-01 and No. 2022GH 007, the One Belt One Road Innovative Talent Exchange with Foreign Experts under Grant No.

DL2022024004L, and the Jinan Scientific Research Leader Studio Project under Grant No. 2021GXRC091. Wei Zhang and Meihong Yang are the corresponding authors.

## References

1. Cao, K., Wei, C., Gaidon, A., Arechiga, N., Ma, T.: Learning imbalanced datasets with label-distribution-aware margin loss. Adv. Neural Inf. Process. Syst. **32** (2019)
2. Chen, H.Y., Chao, W.L.: Fedbe: making bayesian model ensemble applicable to federated learning. arXiv preprint arXiv:2009.01974 (2020)
3. Chu, P., Bian, X., Liu, S., Ling, H.: Feature space augmentation for long-tailed data. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12374, pp. 694–710. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58526-6_41
4. Duan, M., Liu, D., Chen, X., Liu, R., Tan, Y., Liang, L.: Self-balancing federated learning with global imbalanced data in mobile systems. IEEE Trans. Parallel Distrib. Syst. **32**(1), 59–71 (2020)
5. Feng, C., Zhong, Y., Huang, W.: Exploring classification equilibrium in long-tailed object detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 3417–3426 (2021)
6. Hao, H., Xu, C., Zhang, W., Yang, S., Muntean, G.M.: Computing offloading with fairness guarantee: a deep reinforcement learning method. IEEE Trans. Circ. Syst. Video Technol. **33**, 6117–6130 (2023)
7. Hao, H., Xu, C., Zhong, L., Muntean, G.M.: A multi-update deep reinforcement learning algorithm for edge computing service offloading. In: Proceedings of the 28th ACM International Conference on Multimedia, pp. 3256–3264 (2020)
8. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
9. Hinton, G., Vinyals, O., Dean, J.: Distilling the knowledge in a neural network. arXiv preprint arXiv:1503.02531 (2015)
10. Huang, G., Liu, Z., Van Der Maaten, L., Weinberger, K.Q.: Densely connected convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4700–4708 (2017)
11. Jeong, E., Oh, S., Kim, H., Park, J., Bennis, M., Kim, S.L.: Communication-efficient on-device machine learning: federated distillation and augmentation under non-iid private data. arXiv preprint arXiv:1811.11479 (2018)
12. Jiang, C.M., Najibi, M., Qi, C.R., Zhou, Y., Anguelov, D.: Improving the intra-class long-tail in 3d detection via rare example mining. In: Avidan, S., Brostow, G., Cisse, M., Farinella, G.M., Hassner, T. (eds.) ECCV 2022. LNCS, vol. 13670D, pp. 158–175. Springer, Heidelberg (2022). https://doi.org/10.1007/978-3-031-20080-9_10
13. Kang, B., Li, Y., Xie, S., Yuan, Z., Feng, J.: Exploring balanced feature spaces for representation learning. In: International Conference on Learning Representations (2020)
14. Kang, Bet al.: Decoupling representation and classifier for long-tailed recognition. arXiv preprint arXiv:1910.09217 (2019)
15. Karimireddy, S.P., Kale, S., Mohri, M., Reddi, S., Stich, S., Suresh, A.T.: Scaffold: stochastic controlled averaging for federated learning. In: International Conference on Machine Learning, pp. 5132–5143. PMLR (2020)

16. Krizhevsky, A., Hinton, G., et al.: Learning multiple layers of features from tiny images (2009)
17. Li, D., Wang, J.: Fedmd: heterogenous federated learning via model distillation. arXiv preprint arXiv:1910.03581 (2019)
18. Li, T., Sahu, A.K., Talwalkar, A., Smith, V.: Federated learning: challenges, methods, and future directions. IEEE Signal Process. Mag. **37**(3), 50–60 (2020)
19. Li, T., Sahu, A.K., Zaheer, M., Sanjabi, M., Talwalkar, A., Smith, V.: Federated optimization in heterogeneous networks. Proc. Mach. Learn. Syst. **2**, 429–450 (2020)
20. Li, Y., Wang, T., Kang, B., Tang, S., Wang, C., Li, J., Feng, J.: Overcoming classifier imbalance for long-tail object detection with balanced group softmax. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 10991–11000 (2020)
21. Lin, T., Kong, L., Stich, S.U., Jaggi, M.: Ensemble distillation for robust model fusion in federated learning. Adv. Neural. Inf. Process. Syst. **33**, 2351–2363 (2020)
22. Liu, B., Li, H., Kang, H., Hua, G., Vasconcelos, N.: Gistnet: a geometric structure transfer network for long-tailed recognition. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 8209–8218 (2021)
23. Liu, T., Xia, J., Ling, Z., Fu, X., Yu, S., Chen, M.: Efficient federated learning for aiot applications using knowledge distillation. IEEE Internet Things J. (2022)
24. McMahan, B., Moore, E., Ramage, D., Hampson, S., Arcas, B.A.: Communication-efficient learning of deep networks from decentralized data. In: Artificial Intelligence and Statistics, pp. 1273–1282. PMLR (2017)
25. Ren, J., Yu, C., Ma, X., Zhao, H., Yi, S., et al.: Balanced meta-softmax for long-tailed visual recognition. Adv. Neural. Inf. Process. Syst. **33**, 4175–4186 (2020)
26. Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L.C.: Mobilenetv2: inverted residuals and linear bottlenecks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4510–4520 (2018)
27. Sarkar, D., Narang, A., Rai, S.: Fed-focal loss for imbalanced data classification in federated learning. arXiv preprint arXiv:2011.06283 (2020)
28. Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D.: Grad-cam: visual explanations from deep networks via gradient-based localization. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 618–626 (2017)
29. Shang, X., Lu, Y., Huang, G., Wang, H.: Federated learning on heterogeneous and long-tailed data via classifier re-training with federated features. arXiv preprint arXiv:2204.13399 (2022)
30. Sheller, M.J., et al.: Federated learning in medicine facilitating multi-institutional collaborations without sharing patient data. Sci. Rep. **10**(1), 1–12 (2020)
31. Shuai, X., Shen, Y., Jiang, S., Zhao, Z., Yan, Z., Xing, G.: Balancefl: addressing class imbalance in long-tail federated learning. In: 2022 21st ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN), pp. 271–284. IEEE (2022)
32. Dinh, T.C., Tran, N., Nguyen, J.: Personalized federated learning with moreau envelopes. Adv. Neural Inf. Process. Syst. **33**, 21394–21405 (2020)
33. Tan, M., Le, Q.: Efficientnet: rethinking model scaling for convolutional neural networks. In: International Conference on Machine Learning, pp. 6105–6114. PMLR (2019)
34. Wang, H., Yurochkin, M., Sun, Y., Papailiopoulos, D., Khazaeni, Y.: Federated learning with matched averaging. arXiv preprint arXiv:2002.06440 (2020)

35. Wang, J., Liu, Q., Liang, H., Joshi, G., Poor, H.V.: Tackling the objective inconsistency problem in heterogeneous federated optimization. Adv. Neural. Inf. Process. Syst. **33**, 7611–7623 (2020)
36. Wang, L., Xu, S., Wang, X., Zhu, Q.: Addressing class imbalance in federated learning. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, pp. 10165–10173 (2021)
37. Zhang, Y., Kang, B., Hooi, B., Yan, S., Feng, J.: Deep long-tailed learning: a survey. IEEE Trans. Pattern Anal. Mach. Intell. **45**, 10795–10816 (2023)
38. Zheng, W., Yan, L., Gou, C., Wang, F.Y.: Federated meta-learning for fraudulent credit card detection. In: Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence, pp. 4654–4660 (2021)
39. Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., Torralba, A.: Learning deep features for discriminative localization. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2921–2929 (2016)
40. Zhou, B., Cui, Q., Wei, X.S., Chen, Z.M.: BBN: bilateral-branch network with cumulative learning for long-tailed visual recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 9719–9728 (2020)