

Article

Dynamic Client Selection and Group-Balanced Personalization for Data-Imbalanced Federated Speech Recognition

Chundong Xu ¹, Ziyu Wu ¹, Fengpei Ge ^{2,*} and Yuheng Zhi ¹

¹ Jiangxi Provincial Key Laboratory of Multidimensional Intelligent Perception and Control, Jiangxi University of Science and Technology, Ganzhou 341000, China; xuchundong@jxust.edu.cn (C.X.); 6120220316@mail.jxust.edu.cn (Z.W.); 6120240808@mail.jxust.edu.cn (Y.Z.)

² Library, Beijing University of Posts and Telecommunications, Beijing 100876, China

* Correspondence: gefengpei@bupt.edu.cn

Abstract: Federated learning has been widely applied in automatic speech recognition. However, variations in speaker behaviors result in a significant data imbalance across client devices. Conventional federated speech recognition algorithms typically use fixed probabilities to select clients for each round in model training, often overlooking the disparities in data volume among clients. In reality, the substantial differences in data quantity can extend the training duration and compromise the stability of the global model. Moreover, models trained through federated learning on global data often fail to achieve optimal performance for individual local clients. While personalized federated learning strategies hold promise for enhancing model performance, the inherent diversity of speech data makes it challenging to apply state-of-the-art personalized methods effectively to speech recognition tasks. In this paper, a dynamic client selection algorithm is proposed to solve the problem of data disparities among different clients. It can be effectively combined with most federated learning algorithms and dynamically adjusts the selection probabilities of clients based on their dataset size during training. Experimental results demonstrate that this algorithm saved training time by 26% compared to traditional methods on public datasets while maintaining the equivalent model performance. To optimize the personalized federated learning, this paper proposes a novel group-balanced personalization strategy that fine-tunes groupings of clients based on their dataset size. The experimental results show that this algorithm brought a relatively 12% reduction in character error rate, while it did not increase computational costs. In particular, the group-balanced personalization effectively improved the model performance for clients with smaller datasets than local fine-tuning. The combination of dynamic client selection and group-balanced personalization significantly enhanced training efficiency and model performance.



Academic Editor: Yuankai Qi

Received: 12 March 2025

Revised: 2 April 2025

Accepted: 6 April 2025

Published: 7 April 2025

Citation: Xu, C.; Wu, Z.; Ge, F.; Zhi, Y. Dynamic Client Selection and Group-Balanced Personalization for Data-Imbalanced Federated Speech Recognition. *Electronics* **2025**, *14*, 1485. <https://doi.org/10.3390/electronics14071485>

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: speech recognition; federated learning; data-imbalanced; dynamic client selection; group-balanced personalization

1. Introduction

Automatic speech recognition (ASR) is a critical component of artificial intelligence that converts human speech into text. ASR systems based on deep learning have achieved remarkable performance, but they require collecting and preparing vast amounts of training data [1–3]. These data usually come from different sources and need to be stored in a unified processing server or cluster for model optimization. However, speech data often include sensitive personal information. Traditional data collection methods pose a potential risk as they often involve transferring private data among entities [4–6]. This

operation violates regulations such as the General Data Protection Regulation (GDPR), the California Consumer Privacy Act (CCPA), and the Personal Information Protection Law of the People's Republic of China (PIPL). It leads to a large amount of speech data existing in 'data islands' [7]. This situation poses a critical challenge to the collaborative training of ASR models across 'islands'.

To address this challenge, researchers have proposed utilizing federated learning for speech recognition tasks [8–13]. Federated learning is a privacy-preserving distributed machine learning paradigm. It was initially introduced by Google to solve the problem of model updates on Android devices [14]. This approach allows clients to use their own datasets to train a shared model with the assistance of a central server. The training process is performed locally for each client, and data are stored locally to ensure privacy. The work in [8] focused on the development of end-to-end speech recognition models using federated learning, proposing novel aggregation strategies for improved model performance. In [11], Kan et al. explored parameter-efficient transfer learning within the federated learning framework to enhance ASR model performance. Ni et al. [12] proposed a quantized training framework within federated speech recognition to reduce memory usage. Despite federated learning progressing rapidly and being widely applied in common internet scenarios [15,16], it still faces many serious challenges. One of them is the non-independent and identically distributed (non-IID) data. Non-IID refers to the inevitable differences in local data from clients participating in federated training, which include feature distribution skew, label distribution skew, and quantity skew. In federated speech recognition tasks, non-IID is caused by differences in acoustic features and imbalances in data volume [17].

In recent years, many studies have been committed to mitigating the non-IID characteristics of speech data. The work in [18] mitigated client drift by adding Gaussian noise to model parameters during local optimization on the client side. In [19], Zhu et al. proposed two personalized methods to address non-IID in federated speech recognition. Nugyen et al. [20] addressed the non-IID problem of speech data by introducing a self-supervised learning algorithm. Despite the challenges posed by non-IID data, federated learning has been demonstrated to achieve similar accuracy to centralized training in speech recognition tasks [18,19]. However, they primarily focus on differences in the acoustic features but overlook imbalances in data volume. In fact, there can be considerable variation in speaker behaviors. For instance, some individuals frequently use voice assistants, while others rarely do. It leads to huge imbalances in data volume among different client devices. It is critical to addressing the data-imbalance problem in federated learning of speech recognition [21]. Additionally, traditional federated learning algorithms, such as FedAvg [22] and FedProx [23], typically use fixed probabilities to select clients for each round in model training. They do not account for differences in data volume among clients adequately. Such serious differences in data volume can prolong training time and weaken the stability of the global model [24,25].

Moreover, the global model makes it difficult to capture the characteristics of each local dataset [26]. Personalized federated learning has been proposed to solve this problem. The state-of-the-art methods typically employ clustering based on the distribution characteristics of client data [27,28]. Ghosh et al. [29] proposed an iterative clustering framework that assigned clients to different clusters and trains separate models for each cluster. In [30], Huang et al. introduced a personalized cross-silo federated learning approach that used client embeddings to identify and group clients with similar data distributions. However, they were usually applied to image data and rarely to speech data. This is primarily because clustering clients by speech dataset similarity is challenging due to variations in accents, speech rates, and content, and would incur high computational costs [31]. In fact, local

fine-tuning remains the default personalization method in federated speech recognition, but it provides limited benefits, particularly for clients with smaller datasets.

This paper proposes a dynamic client selection algorithm and a group-balanced personalization strategy to address these challenges. The federated speech recognition system was decoupled into two stages: (1) In the global model training stage, the probabilities of client selection were adjusted through a segmented function. Initially, clients with smaller datasets were prioritized to accelerate the learning of their unique features and mitigate the overemphasis on clients with larger datasets. As the training progressed, the probabilities were adjusted to give higher selection priorities to clients with larger datasets until convergence. This process utilized the extensive data to optimize the model and improve its robustness. (2) In the personalization training stage, the group-balanced personalization strategy was introduced. This approach began with categorizing clients according to their dataset size. Following this, varying rounds of federated fine-tuning were applied to each group. The process concluded with local fine-tuning to further tailor the model to individual client needs. On the one hand, the grouping approach mitigated differences in dataset size among clients within each group without incurring heavy computational costs. On the other hand, different rounds of federated fine-tuning balanced the underrepresentation of clients with smaller datasets that occurred in the later phase of global model training. It also mitigated the limitations of traditional local fine-tuning for clients with smaller datasets.

The rest of this article is organized as follows: Section 2 introduces the architecture of the automatic speech recognition model and associated federated learning algorithms. In Section 3, we present a detailed discussion of dynamic client selection and group-balanced personalization. The datasets and evaluation metrics are described in Section 4. Section 5 provides an in-depth analysis of the experimental results. Lastly, we summarize the key points of this paper.

2. Related Foundations

2.1. Architecture of Speech Recognition Model

This work employed an open-source framework Automatic Speech Recognition Toolkit [32] to construct an end-to-end Deep Convolutional Neural Networks—Connectionist Temporal Classification (DCNN-CTC) model for experiments [5,33]. The model architecture was inspired by the classic VGG network configuration [34]. It is distinguished by its simplicity, lower parameter count, rapid convergence, and strong scalability. Its depth can be adjusted by changing the number of convolutional layers to match different training data volumes. These characteristics make it particularly well suited to our research.

The architecture of this model is illustrated in Figure 1. Initially, feature extraction is applied to the input data, generating a 200-dimensional spectrogram. Subsequently, multiple convolutional and pooling layers are applied to extract high-level acoustic features, with each convolutional layer being accompanied by batch normalization and ReLU activation. Afterward, a reshape layer transforms these deep feature sequences into a 3D tensor, preparing them for fully connected layers. This is followed by two fully connected layers. The first layer employs a ReLU activation function, while the second layer uses a softmax activation function to generate a probability distribution for the output. The CTC loss function is applied to optimize the model. The details of the proposed model are listed in Table 1.

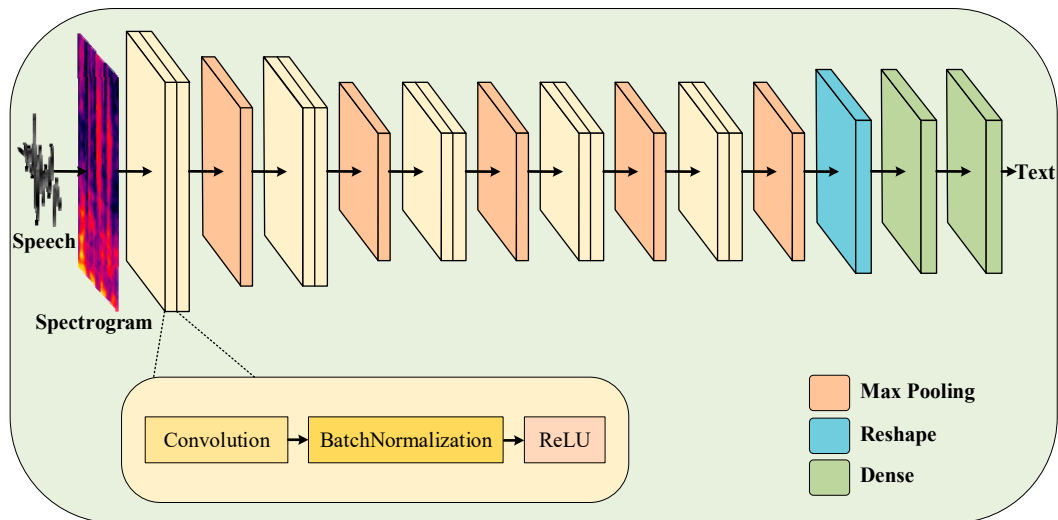


Figure 1. DCNN-CTC model architecture.

Table 1. Layers of the DCNN-CTC model.

Layer No.	Type of Layer	Kernel Size	Pooling Size	Step Size	Number of Neurons
1	Conv2D	(3×3)	-	1	32
2	MaxPooling	-	(2×2)	2	32
3	Conv2D	(3×3)	-	1	64
4	MaxPooling	-	(2×2)	2	64
5	Conv2D	(3×3)	-	1	128
6	MaxPooling	-	(2×2)	2	128
7	Conv2D	(3×3)	-	1	128
8	MaxPooling	-	(1×1)	2	128
9	Conv2D	(3×3)	-	1	128
10	MaxPooling	-	(1×1)	2	128
11	Reshape	-	-	-	256
12	Dense	-	-	-	128
13	Dense	-	-	-	1428

2.2. Associated Federated Learning Algorithms

The existing research on global model training of federated speech recognition predominantly follows the FedAvg [22] paradigm. The framework for this process is depicted in Figure 2 and consists of five key steps:

- (1) The model is pre-trained on the server using a centralized method for parameter initialization.
- (2) The server randomly selects M clients from the entire pool of clients using fixed probabilities and distributes the model parameters to them.
- (3) Clients utilize local data to train the parameters and subsequently return the updated model parameters to the server.
- (4) The server collects updated model parameters from participating clients, then performs weighted aggregation and update global model parameters. The global model parameter is given as:

$$F(\omega) = \sum_{k=1}^M \frac{D_k}{D} f_k(\omega) \quad (1)$$

where $f_k(\omega)$ is the local model parameters of client k , D_k is the dataset size of client k , and D is the total dataset size of M clients.

- (5) Repeat steps 2–4 until the global model converges, then distribute the final model parameters to all clients.

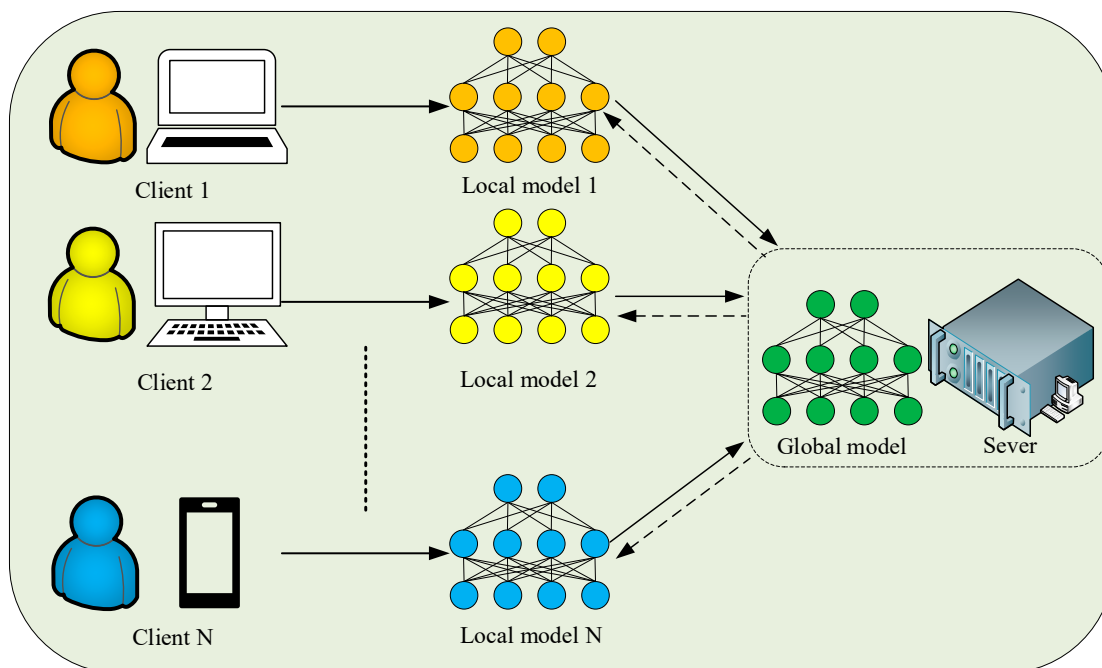


Figure 2. Framework of federated speech recognition.

FedAvg is characterized by its straightforward structure and ease of implementation. Its performance and reliability have been extensively validated across diverse scenarios and datasets. Despite being proposed earlier, it continues to offer significant research potential, especially in speech recognition tasks. As a result, it remains a benchmark for comparison in recent studies on federated speech recognition [8,9,11–13,18–20]. However, FedAvg faces challenges when dealing with non-IID (e.g., speech samples with different speakers or data volume). These limitations motivated the development of FedProx [23].

FedProx is one of the most widely utilized improvements of the FedAvg that maintains the advantages of FedAvg while addressing its key limitations in heterogeneous settings. It achieves state-of-the-art performance on some heterogeneous computer vision scenarios. In FedProx a regularization term is introduced to the loss function in training the local model on the client, which helps parameter updates not deviate substantially from the global model and effectively enhances system performance in the scenarios with the non-IID problem. The objective function for local training is as follows:

$$L = L_{loss} + \frac{\mu}{2} \|\omega - \omega_{global}\|^2 \quad (2)$$

where L_{loss} is the original loss function, μ is an adjustable parameter that controls the weight of the regularization term, ω denotes the parameters of the local model, and ω_{global} represents the parameters of the global model. Additionally, FedProx determines client selection probabilities in proportion to their local dataset sizes, prioritizing those with larger data volumes. Moreover, the global model is aggregated by direct averaging without assigning specific weights. In this work, FedProx was utilized as one of the baselines.

3. Methods

3.1. Dynamic Client Selection

The dynamic client selection algorithm was inspired by a widely used training strategy in machine learning. This strategy involves initially training the model with a smaller or simpler subset of the dataset to quickly establish the foundational structure of the model, followed by comprehensive training with a larger or more complex dataset to improve the performance and robustness of the model. This idea is typically utilized in curriculum learning especially in big data settings [35,36]. This work builds on the concept and introduces the dynamic client selection algorithm. Figure 3 illustrates the federated training process combined with this algorithm.

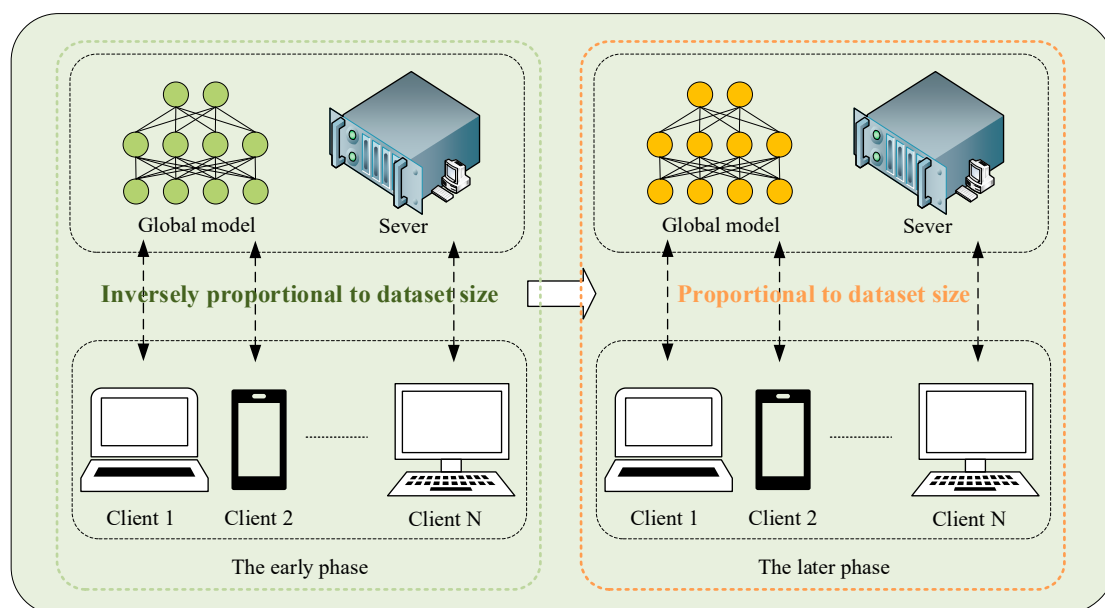


Figure 3. Federated training with dynamic client selection.

Specifically, we considered a practical setting of data-imbalanced federated speech recognition involving N clients. Each client k had a local dataset of size D_k , and the probability of being selected in the current training round was P_k . The global model training stage was divided into two phases through the dynamic client selection algorithm:

- (1) The early phase. Clients with smaller datasets were prioritized with a higher probability of being selected for training. This approach aimed to leverage smaller datasets for rapid initial fitting, thereby minimizing time loss. It also mitigated the disproportionate influence of clients with larger datasets on the global model. In this work, the selection probability for client k during this phase was inversely proportional to dataset size across clients.
- (2) The later phase. After a certain number of rounds, this algorithm adjusted the strategy to give clients with larger datasets higher priority until the model converges. This approach ensured that more comprehensive data contributed to fine-tuning the global model and enhancing robustness. In this phase, the selection probability for client k was proportional to dataset size.

Furthermore, the algorithm needs to know all potential clients before the training begins. This allows it to handle clients joining late or dropping out early. New clients can only join after the current training round finishes, while active clients that encounter failures or choose to exit can withdraw immediately. The server will recalculate the client selection probabilities for the remaining clients after the current round completes.

We utilized a piecewise function to dynamically adjust the selection probabilities, where r is the current training round of the global model, and R is the threshold for switching probability.

$$P_k(r) = \begin{cases} \frac{1/D_k}{\sum_{k=1}^N 1/D_k}, & (0 \leq r < R) \\ \frac{D_k}{\sum_{k=1}^N D_k}, & (R \leq r < \text{Convergence}) \end{cases} \quad (3)$$

It is crucial to emphasize that determining the threshold R in this algorithm is important. Based on prior experience, the threshold R is generally set at the point of initial convergence during the early training phase. However, this approach faces implementation challenges in practical deployments. Two key limitations require systematic analysis:

- (1) The distribution of client data. If there are serious differences in the distributions of client data, it may hinder the initial convergence of the global model during the early training phase. In such cases, it is necessary to determine an appropriate R based on the specific structure of the model and its convergence speed under centralized training conditions. If these conditions are difficult to satisfy, R can be set when the value of the loss function remains non-decreasing for a certain number of rounds.
- (2) Resource constraints. If computational resources are limited, it might be necessary to adjust probabilities before the initial convergence of the model to conserve resources. However, this approach may prevent the global model from fully learning the characteristics in training data, which could ultimately compromise its recognition performance. In such cases, it is essential to balance training costs and model performance to determine an appropriate R .

3.2. Group-Balanced Personalization

The strategy of group-balanced personalization is based on two key principles. First, due to the influences of the dynamic client selection algorithm, clients with smaller datasets have a lower chance of being selected during the later phase of global model training. Although these clients receive sufficient training in the early phase, the catastrophic forgetting of deep learning causes the adaptability of the model to their data to decline as training continues [37]. Ultimately, it impacts the performance of the personalized models for these clients. Second, state-of-the-art federated personalized methods typically employ clustering based on the distribution characteristics of client data [27–30]. The application of these methods in speech recognition involves considerable challenges. Clustering clients based on speech datasets is related to speaker recognition tasks. Applying these methods in real-world speech recognition scenarios result in heavy computational costs [29]. In addition, traditional local fine-tuning performs poorly for clients with smaller datasets.

As a result, this work aims to perform grouping fine-tuning based on data volume. Despite not precisely matching clients with similar data characteristics, this method helps mitigate the huge gap in data volume among clients within each group. Moreover, this approach avoids heavy computational costs. Particularly, we can reduce computational costs by adjusting the number of fine-tuning rounds. The specific method is as follows:

To address differences in dataset size, we initially grouped clients by clustering those with similar dataset size. This approach enhanced the stability of subsequent training without incurring the heavy computational overhead. Subsequently, federated fine-tuning was applied independently within each group at varying degrees. Groups consisting of clients with smaller datasets underwent more thorough fine-tuning, while those with larger datasets required fewer rounds of fine-tuning. This approach balanced the underrepresentation of clients with smaller datasets in the later phase of global model training, ensuring

the full utilization of all client data. Finally, building on the group-balanced federated fine-tuning, local fine-tuning was performed using the specific data of individual clients. This enhanced the model adaptation to the unique data distribution of each client. The proposed group-balanced personalization is presented in Algorithm 1 to clearly illustrate the aforementioned process. Figure 4 shows the personalized training process.

Algorithm 1: Proposed group-balanced personalization. K is the number of groups, M represents models, N represents the number of clients.

Phase 1 (Group-Based Federated Fine-Tuning):

Extract K groups by clustering clients with similar dataset size

for each group $g \in K$ **do**

if g consists of clients with smaller datasets **do**

 fully federated fine-tuning within g

else if g consists of clients with larger datasets **do**

 a small amount of federated fine-tuning or no fine-tuning within g

end for

Return $(M_1, M_2, \dots, M_K)$

Phase 2 (Local Fine-Tuning):

for each client $c \in N$ **do**

if c within g **do**

 utilize the local dataset in c to fully fine-tune M_g

end for

Return $(M^1, M^2, \dots, M^N)$

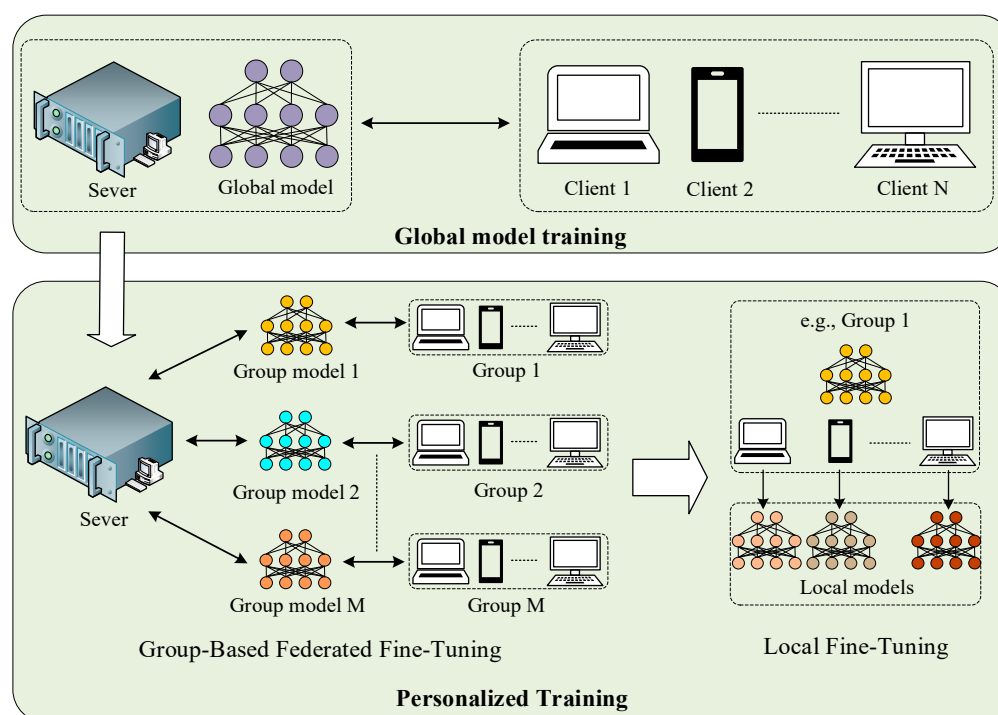


Figure 4. Personalized training process.

4. Experimental Setup

4.1. System Configuration

We conducted the federated learning environment on a server equipped with an NVIDIA GeForce RTX 2080 Ti GPU (11 GB VRAM), running on Ubuntu 20.04 LTS. The

development environment was configured with Python 3.9 and PyCharm IDE, with 1 TB of storage allocated for data processing and model checkpoints.

4.2. Datasets

In order to validate the effectiveness of the proposed algorithms, three datasets were utilized, which were THCHS30 [38], ST-CMDS [39], and AISHELL-1 [40]. These datasets consist of speech recordings from speakers across various regions of China. Respectively, THCHS30 and ST-CMDS contain 25 and 100 h of training data, both utilized for initializing the global model. The AISHELL-1 includes a training set of 150 h speech from 340 different speakers, a development set of 18 h speech from 40 different speakers, and a test set of 10 h speech from 20 different speakers. AISHELL-1 was further used to create a data-imbalanced federated speech recognition environment. The data allocation followed three key criteria:

- (1) Diversity in acoustic features across clients: Typically, the speakers associated with different clients were distinct, and these speakers often exhibited considerable variations in speech rate, intonation, and other characteristics.
- (2) Imbalance in dataset size across clients: As different clients serviced distinct speakers with varying speaking habits, there tended to be a noticeable imbalance in the amount of speech data across various clients.
- (3) Evaluation of personalized local models on client-specific datasets: The performance of each personalized model was evaluated using the unique dataset of each client.

The detailed data distribution is shown in Table 2. Specifically, we first utilized the Development and Test sets from AISHELL-1 as the Server Development and Test sets to evaluate the global model performance. Subsequently, the Training set of AISHELL-1 was randomly divided into 10 groups based on different speakers, with each group serving as the local data for Client0-9. Among these, 5 groups each included data from 20 speakers, 3 groups each included data from 40 speakers, and 2 groups each included data from 60 speakers. Finally, 10 sets of data were randomly selected from the local datasets of Client0 to Client9, with 20 utterances per speaker. These sets served as the personalized test sets for each client to evaluate local model performance. The remaining local data were used as local training data.

Table 2. The detailed data distribution.

Server Data Distribution			
Server	Speakers	Development Duration (hours)	Test Duration (hours)
	60	18	10
Client Data Distribution			
Client ID	Speakers	Train Duration (hours)	Test Duration (hours)
0	20	8.13	1.04
1	20	7.36	0.97
2	20	7.19	0.96
3	20	7.76	0.98
4	20	7.52	0.97
5	40	15.57	1.97
6	40	15.88	2.02
7	40	15.72	2.00
8	60	24.26	3.05
9	60	24.44	3.06

It is important to highlight that this study focuses on local data volume differences across clients, but our experimental design explicitly accounts for speaker variability.

Specifically, each client was assigned unique speakers with distinct acoustic characterized (e.g., accent, age, and gender).

4.3. Evaluation Metrics

Character Error Rate (CER) is utilized as the evaluation metric for the speech recognition model in our study. The formula for calculating CER is given as:

$$CER = \frac{S + D + I}{N} \quad (4)$$

S , D , and I represent the number of substitutions, deletions, and insertions, respectively, and N represents the total number of characters in the reference sequence.

5. Experiment and Comparison

In this section, three sets of experiments are designed to evaluate the effectiveness of the dynamic client selection algorithm and the group-balanced personalization strategy. Initially, dynamic client selection was combined with FedAvg, and experiments were conducted with different values of R to analyze how threshold selection impacts both model performance and training time. Subsequently, dynamic client selection was combined with FedProx for a comprehensive examination of algorithm performance. Finally, two global models were selected from the previous experiments for the personalization experiments.

5.1. Dynamic Client Selection with FedAvg

FedAvg was used as the baseline in this set of experiments. In each round, three clients were selected by the server, and each client performed one epoch of local training. The global model CER variations under different R are shown in Figures 5–10.

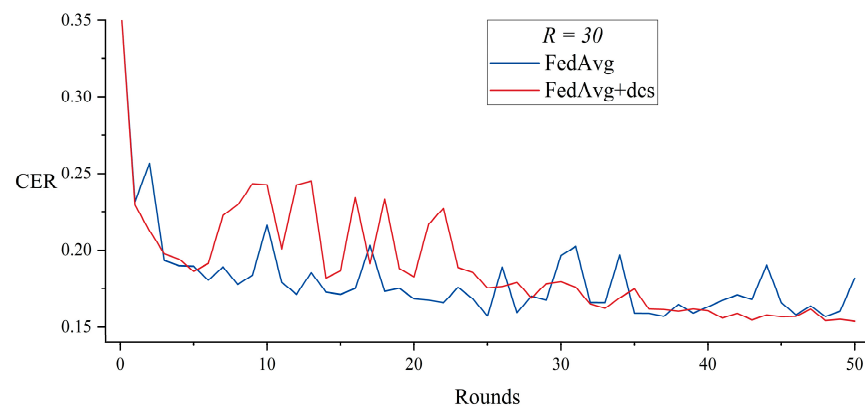


Figure 5. The CERs on the server dev ($R = 30$).

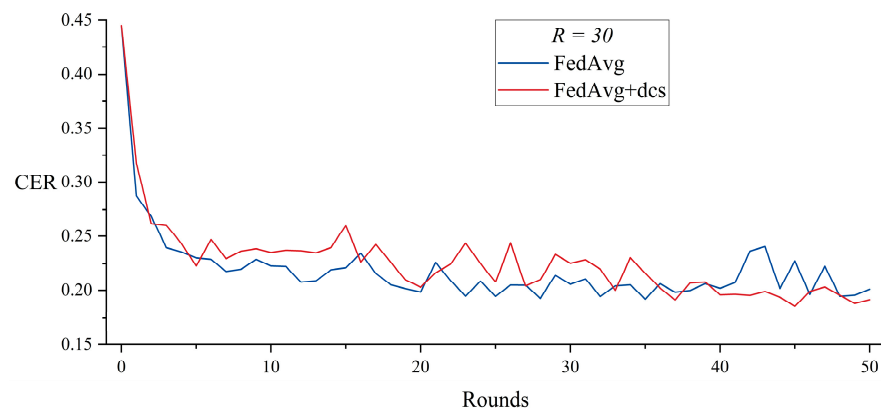


Figure 6. The CERs on the server test ($R = 30$).

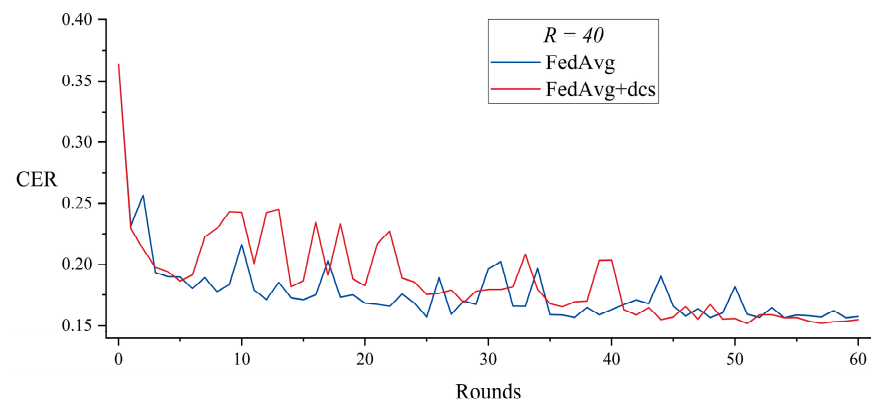


Figure 7. The CERs on the server dev ($R = 40$).

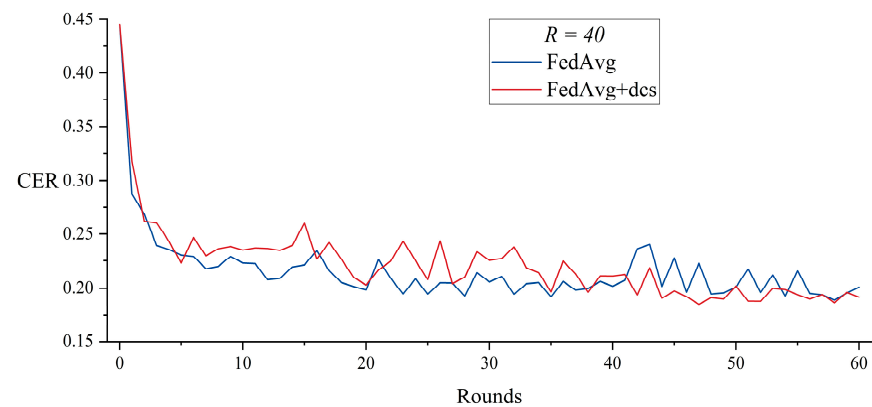


Figure 8. The CERs on the server test ($R = 40$).

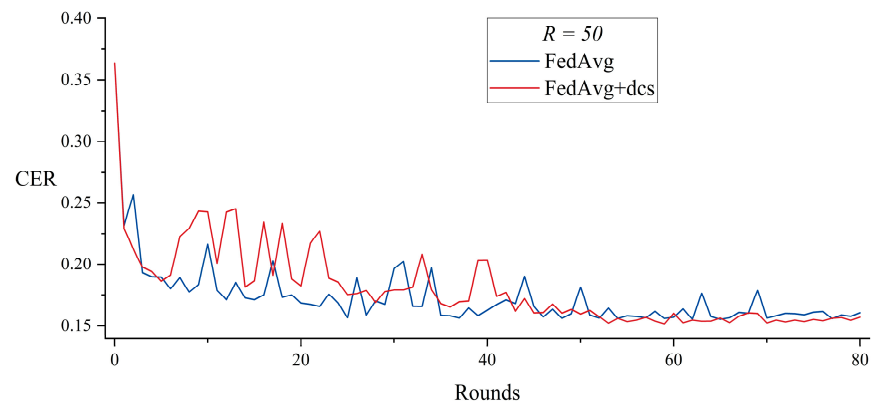


Figure 9. The CERs on the server dev ($R = 50$).

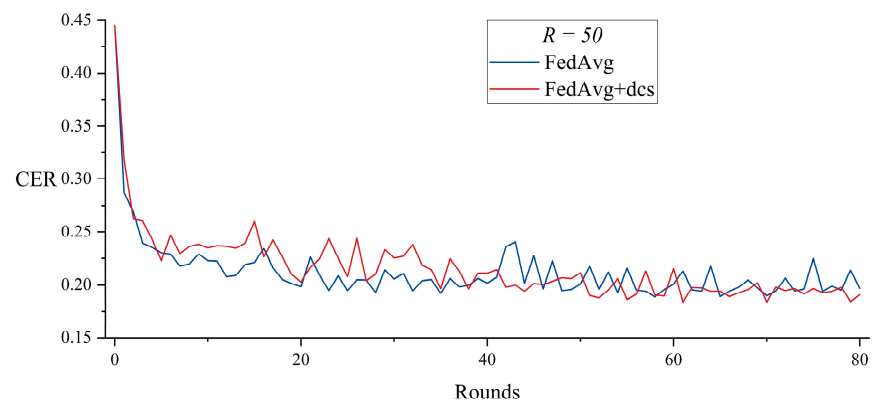


Figure 10. The CERs on the server test ($R = 50$).

As shown in Figures 5–10, data-imbalanced scenarios resulted in fluctuations in the performance of FedAvg and delayed its convergence. This is because FedAvg does not account for imbalances in client data volume when selecting clients for training. Consequently, the global model was incapable of effectively capturing the distinct data distributions of each client. However, the implementation of a dynamic client selection algorithm resulted in notable early fluctuations, which were more pronounced than those observed with FedAvg. After adjusting the selection probabilities, the model converged within a few rounds, and the training process became more stable compared to FedAvg.

The dynamic client selection algorithm selects clients for training according to different criteria at different times, fully utilizing the imbalances in volume of client data. In the early phase of global model training, the dynamic client selection algorithm prioritizes clients with smaller datasets. While this approach promotes initial exploration of diverse data patterns, the limited samples from these clients cannot sufficiently represent the overall data distribution, resulting in unstable parameter updates that manifest as performance fluctuations. After adjusting the selection probabilities, the server frequently selects clients with larger datasets. As these clients provide statistically dominant data samples, the model progressively aligns with the principal data characteristics. This strategic shift builds upon the preliminary understanding of minority patterns acquired during early-stage training, enabling the global model to achieve accelerated convergence while maintaining stability.

Moreover, our results indicate that increasing the value of R could improve the performance of the global model to a certain extent. However, this improvement comes at the cost of requiring more rounds to approach convergence, and the gains in performance are ultimately limited.

To further analyze the performance of the algorithm, the training times of the global model for different algorithms were recorded. The results are shown in Table 3. It was determined that implementing a dynamic client selection algorithm significantly reduced the model training time. Specifically, the training time reduction was 28.37% for $R = 30$, 27.08% for $R = 40$, and 24.07% for $R = 50$. Additionally, a larger R value increased the model convergence time. Consequently, the employment of dynamic client selection necessitated a right balance between the duration of training and the efficacy of the model.

Table 3. Comparison of global model training duration (FedAvg baseline). bold values highlight the shortest training duration, N/A indicates non-convergence.

Setting	Method	Total Duration (s)	Convergence	
			Rounds	Duration (s)
$R = 30$, Rounds = 50	FedAvg	30,146	N/A	N/A
	FedAvg + dcs	26,586	43	21,591
$R = 40$, Rounds = 60	FedAvg	35,939	N/A	N/A
	FedAvg + dcs	31,545	52	26,207
$R = 50$, Rounds = 80	FedAvg	46,850	N/A	N/A
	FedAvg + dcs	43,469	68	35,597

5.2. Dynamic Client Selection with FedProx

To further analyze the performance of the dynamic client selection, we conducted experiments utilizing FedProx as the baseline. In each round, the server selected three clients, and each client completed one epoch of local training, with R set to 30. Figures 11–14 and Table 4 show the results obtained with different values of the adjustable parameter μ .

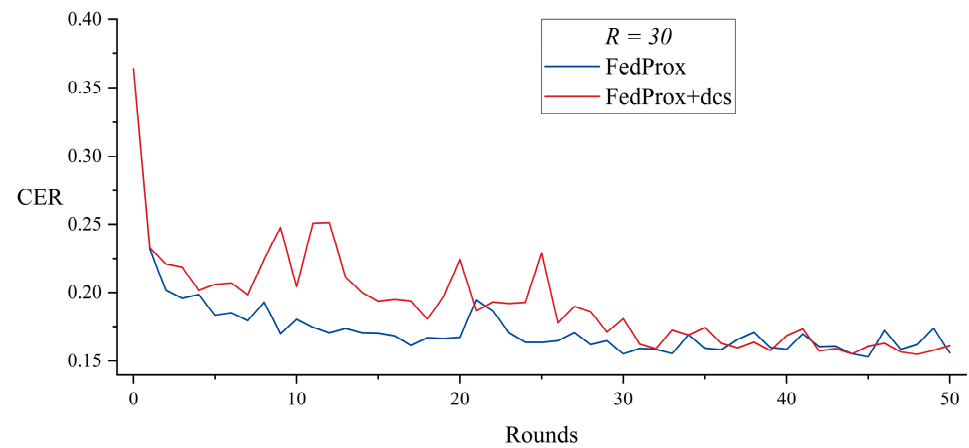


Figure 11. The CERs on the server dev ($\mu = 0.1$).

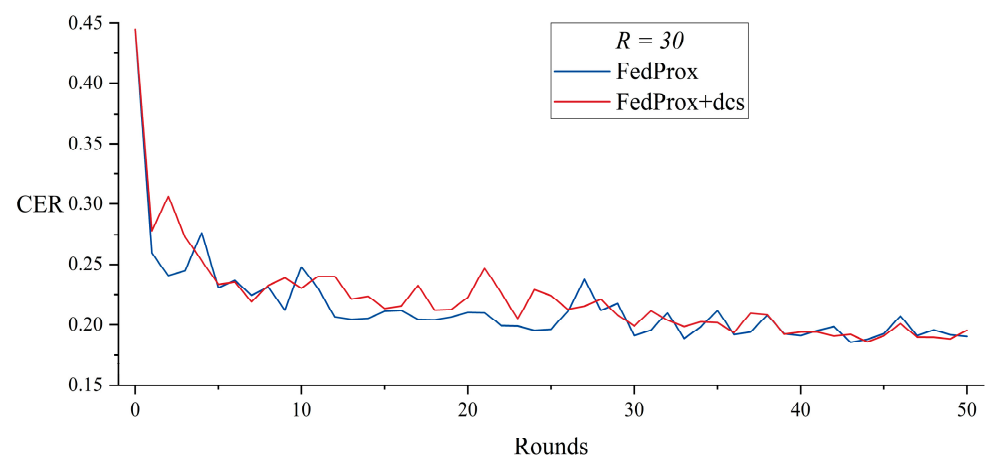


Figure 12. The CERs on the server test ($\mu = 0.1$).

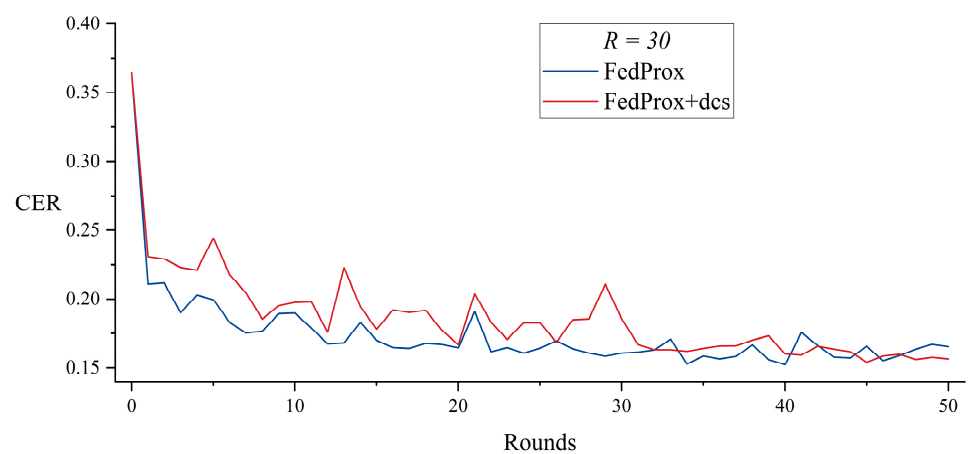


Figure 13. The CERs on the server dev ($\mu = 0.01$).

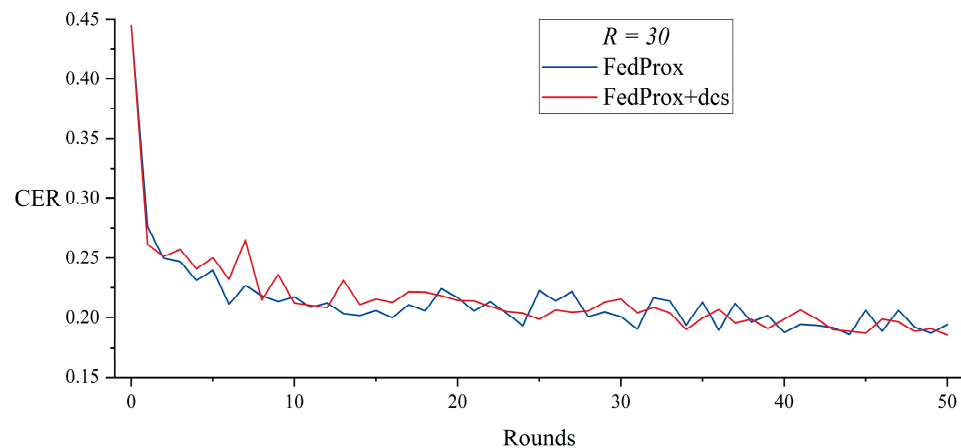


Figure 14. The CERs on the server test ($\mu = 0.01$).

Table 4. Comparison of global model training duration. bold values highlight the shortest training duration.

Setting	Method	Total Duration (s)	Convergence	
			Rounds	Duration (s)
$R = 30, \mu = 0.1$	FedProx	38,995	43	31,650
	FedProx + dcs	27,850	43	23,403
$R = 30, \mu = 0.01$	FedProx	36,734	45	31,951
	FedProx + dcs	28,572	43	23,385

As shown in Figures 11–14 and Table 4, while FedProx achieved competitive CERs in data-imbalanced federated speech recognition tasks, combining the dynamic client selection mechanism demonstrated superior time efficiency—reducing training time by 26.06% ($\mu = 0.1$) and 26.81% ($\mu = 0.01$) while maintaining comparable model accuracy. This is because FedProx selects clients with larger datasets for systematic prioritization in training aggregation, which mitigates convergence difficulties caused by data imbalance but incurs high computational overhead. The inherent regularization term in FedProx further compounds this issue, as client computational costs increase scale substantially with model complexity. Our algorithm strategically addresses these limitations through introducing the dynamic client selection algorithm.

5.3. Results of Personalized Model

To verify the effectiveness of the group-balanced personalization strategy, a further experiment was conducted based on the preceding experiments. The CERs on the test sets of each client are presented in Table 5. Specifically, the clients were clustered into three groups using the group-balanced personalization strategy. To facilitate the analysis of the effectiveness of the strategy, the experiment employed five rounds of federated fine-tuning without changing the number of rounds. Subsequently, the group models were further fine-tuned using the local data from the corresponding clients to obtain personalized local models. The results of the group-balanced personalization strategy are labeled as ‘Ours’. The traditional local fine-tuning serves as the baseline for comparison, indicated as ‘Traditional Personalization’. Two global models were selected for the experiments: Server_model 1 was selected from FedAvg with dynamic client selection, and Server_model 2 was selected from FedProx with dynamic client selection. The CER values for both models on the test sets are presented as ‘Initial’.

Table 5. Personalization models CERs. Best CERs are shown in bold.

Client Test ID		Sever_Model 1			Sever_Model 2		
		Initial	Traditional Personalization	Ours	Initial	Traditional Personalization	Ours
Group1	0	17.92%	15.02%	14.56%	16.28%	15.19%	14.92%
	1	18.12%	15.75%	15.45%	17.60%	15.75%	15.48%
	2	17.35%	15.77%	15.32%	17.35%	15.65%	15.57%
	3	19.83%	16.44%	16.51%	19.23%	16.89%	16.63%
	4	18.07%	15.47%	15.27%	18.60%	15.50%	15.29%
Group2	5	14.92%	14.35%	14.38%	15.19%	14.35%	14.51%
	6	13.33%	11.57%	11.46%	14.13%	11.47%	11.47%
	7	13.94%	12.18%	12.02%	14.17%	12.10%	11.97%
Group3	8	18.46%	16.52%	16.36%	18.53%	16.48%	16.55%
	9	18.98%	18.53%	18.51%	19.93%	18.79%	18.59%

As shown in Table 5, the group-balanced personalization strategy effectively improved the performance of local models. An average CER reduction of 12% was achieved compared to the global model. Meanwhile, the performance on clients with smaller datasets was significantly better than the traditional personalization method. Moreover, for clients with larger datasets, grouping for federated fine-tuning could enhance the performance of the final personalized local models, although the improvements were less significant compared to clients with smaller datasets. Consequently, minimal federated fine-tuning was sufficient for groups consisting of clients with larger datasets, and in some instances, this step could even be skipped entirely.

6. Conclusions

This paper addresses the challenges of client selection and personalization in data-imbalanced federated speech recognition, making several key contributions. First, we proposed a dynamic client selection algorithm that mitigated issues of wasted training time and instability commonly seen in traditional methods. Based on this, we introduced a group-balanced personalization strategy that effectively improved the performance of client local models, especially for clients with smaller datasets. The combination of these two approaches ensures an efficient training process and achieves strong model performance. Looking ahead, our future research will focus on developing a more effective federated client selection algorithm for realistic speech recognition. Specifically, we plan to use the complexity of local datasets as one of the criteria for selection. Additionally, we aim to conduct experiments with a larger number of clients to further verify the applicability and scalability of the proposed algorithm.

Author Contributions: Methodology, Z.W.; Validation, Y.Z.; Resources, C.X.; Writing—original draft, Z.W.; Writing—review & editing, F.G.; Supervision, C.X.; Funding acquisition, F.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China (Grant No. 12204062).

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Fatehi, K.; Torres, M.T.; Kucukyilmaz, A. An overview of high-resource automatic speech recognition methods and their empirical evaluation in low-resource environments. *Speech Commun.* **2024**, *167*, 103151.
2. Kheddar, H.; Hemis, M.; Himeur, Y. Automatic speech recognition using advanced deep learning approaches: A survey. *Inf. Fusion* **2024**, *109*, 102422.
3. Zhang, L.; Wu, S.; Wang, Z. End-to-End Speech Recognition with Deep Fusion: Leveraging External Language Models for Low-Resource Scenarios. *Electronics* **2025**, *14*, 802. [\[CrossRef\]](#)
4. Jiang, D.; Tan, C.; Peng, J.; Chen, C.; Wu, X.; Zhao, W.; Song, Y.; Tong, Y.; Liu, C.; Xu, Q.; et al. A GDPR-compliant ecosystem for speech recognition with transfer, federated, and evolutionary learning. *ACM Trans. Intell. Syst. Technol. (TIST)* **2021**, *12*, 1–19.
5. Zhou, Y.; Cui, F.; Che, J.; Ni, M.; Zhang, Z.; Li, J. Elastic Balancing of Communication Efficiency and Performance in Federated Learning with Staged Clustering. *Electronics* **2025**, *14*, 745. [\[CrossRef\]](#)
6. Qi, P.; Chiaro, D.; Guzzo, A.; Ianni, M.; Fortino, G.; Piccialli, F. Model aggregation techniques in federated learning: A comprehensive survey. *Future Gener. Comput. Syst.* **2024**, *150*, 272–293.
7. Nandury, K.; Mohan, A.; Weber, F. Cross-silo federated training in the cloud with diversity scaling and semi-supervised learning. In Proceedings of the ICASSP 2021–2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Toronto, ON, Canada, 6–11 June 2021; pp. 3085–3089.
8. Gao, Y.; Parcollet, T.; Zaiem, S.; Fernandez-Marques, J.; De Gusmao, P.P.; Beutel, D.J.; Lane, N.D. End-to-end speech recognition from federated acoustic models. In Proceedings of the ICASSP 2022–2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Singapore, 23–27 May 2022; pp. 7227–7231.
9. Tsouvalas, V.; Saeed, A.; Ozcelebi, T. Federated self-training for data-efficient audio recognition. In Proceedings of the ICASSP 2022–2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Singapore, 23–27 May 2022; pp. 476–480.
10. Yang, C.H.H.; Chen, I.F.; Stolcke, A.; Siniscalchi, S.M.; Lee, C.H. An experimental study on private aggregation of teacher ensemble learning for end-to-end speech recognition. In Proceedings of the 2022 IEEE Spoken Language Technology Workshop (SLT), Doha, Qatar, 9–12 January 2023; pp. 1074–1080.
11. Kan, X.; Xiao, Y.; Yang, T.; Chen, N.; Mathews, R. Parameter-Efficient Transfer Learning under Federated Learning for Automatic Speech Recognition. *arXiv* **2024**, arXiv:2408.11873.
12. Ni, R.; Xiao, Y.; Meadowlark, P.; Rybakov, O.; Goldstein, T.; Suresh, A.T.; Moreno, I.L.; Chen, M.; Mathews, R. FedAQT: Accurate Quantized Training with Federated Learning. In Proceedings of the ICASSP 2024–2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Seoul, Republic of Korea, 14–19 April 2024; pp. 6100–6104.
13. Du, Y.; Zhang, Z.; Yue, L.; Huang, X.; Zhang, Y.; Xu, T.; Xu, L.; Chen, E. Communication-Efficient Personalized Federated Learning for Speech-to-Text Tasks. In Proceedings of the ICASSP 2024–2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Seoul, Republic of Korea, 14–19 April 2024; pp. 10001–10005.
14. AbdulRahman, S.; Tout, H.; Ould-Slimane, H.; Mourad, A.; Talhi, C.; Guizani, M. A survey on federated learning: The journey from centralized to distributed on-site learning and beyond. *IEEE Internet Things J.* **2020**, *8*, 5476–5497.
15. Zhang, F.; Shuai, Z.; Kuang, K.; Wu, F.; Zhuang, Y.; Xiao, J. Unified fair federated learning for digital healthcare. *Patterns* **2024**, *5*, 100907.
16. Solomon, E.; Woubie, A. Federated Learning Method for Preserving Privacy in Face Recognition System. *arXiv* **2024**, arXiv:2403.05344.
17. Farahani, B.; Tabibian, S.; Ebrahimi, H. Towards a Personalized Clustered Federated Learning: A Speech Recognition Case Study. *IEEE Internet Things J.* **2023**, *10*, 18553–18562.
18. Guliani, D.; Beaufays, F.; Motta, G. Training speech recognition models with federated learning: A quality/cost framework. In Proceedings of the ICASSP 2021–2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Toronto, ON, Canada, 6–11 June 2021; pp. 3080–3084.
19. Zhu, H.; Wang, J.; Cheng, G.; Zhang, P.; Yan, Y. Decoupled federated learning for asr with non-iid data. *arXiv* **2022**, arXiv:2206.09102.
20. Nguyen, T.; Mdhaffar, S.; Tomashenko, N.; Bonastre, J.F.; Estève, Y. Federated learning for asr based on wav2vec 2.0. In Proceedings of the ICASSP 2023–2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Rhodes Island, Greece, 4–10 June 2023; pp. 1–5.
21. Kaur, H.; Rani, V.; Kumar, M.; Sachdeva, M.; Mittal, A.; Kumar, K. Federated learning: A comprehensive review of recent advances and applications. *Multimed. Tools Appl.* **2024**, *83*, 54165–54188.
22. McMahan, B.; Moore, E.; Ramage, D.; Hampson, S.; y Arcas, B.A. Communication-efficient learning of deep networks from decentralized data. In Proceedings of the Artificial Intelligence and Statistics, PMLR, Fort Lauderdale, FL, USA, 20–22 April 2017; pp. 1273–1282.
23. Li, T.; Sahu, A.K.; Zaheer, M.; Sanjabi, M.; Talwalkar, A.; Smith, V. Federated optimization in heterogeneous networks. *Proc. Mach. Learn. Syst.* **2020**, *2*, 429–450.

24. Karimireddy, S.P.; Kale, S.; Mohri, M.; Reddi, S.; Stich, S.; Suresh, A.T. Scaffold: Stochastic controlled averaging for federated learning. In Proceedings of the International Conference on Machine Learning, PMLR, Online, 13–18 July 2020; pp. 5132–5143.
25. Wen, J.; Zhang, Z.; Lan, Y.; Cui, Z.; Cai, J.; Zhang, W. A survey on federated learning: Challenges and applications. *Int. J. Mach. Learn. Cybern.* **2023**, *14*, 513–535.
26. Tan, A.Z.; Yu, H.; Cui, L.; Yang, Q. Towards personalized federated learning. *IEEE Trans. Neural Netw. Learn. Syst.* **2022**, *34*, 9587–9603. [CrossRef]
27. Xu, J.; Tong, X.; Huang, S.L. Personalized federated learning with feature alignment and classifier collaboration. *arXiv* **2023**, arXiv:2306.11867.
28. Lin, I.; Yagan, O.; Joe-Wong, C. FedSPD: A Soft-clustering Approach for Personalized Decentralized Federated Learning. *arXiv* **2024**, arXiv:2410.18862.
29. Ghosh, A.; Chung, J.; Yin, D.; Ramchandran, K. An efficient framework for clustered federated learning. In Proceedings of the 34th International Conference on Neural Information Processing Systems, Vancouver, BC, Canada, 6–12 December 2020; pp. 19586–19597.
30. Huang, Y.; Chu, L.; Zhou, Z.; Wang, L.; Liu, J.; Pei, J.; Zhang, Y. Personalized cross-silo federated learning on non-IID data. *Proc. AAAI Conf. Artif. Intell.* **2021**, *35*, 7865–7873.
31. Bai, Z.; Zhang, X.L. Speaker recognition based on deep learning: An overview. *Neural Netw.* **2021**, *140*, 65–99. [CrossRef]
32. Nl8590687. A Deep-Learning-Based Chinese Speech Recognition System. Available online: https://github.com/nl8590687/ASRT_SpeechRecognition (accessed on 26 July 2024).
33. Dong, Z.; Ding, Q.; Zhai, W.; Zhou, M. A speech recognition method based on domain-specific datasets and confidence decision networks. *Sensors* **2023**, *23*, 6036. [CrossRef] [PubMed]
34. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
35. Karim, N.; Mithun, N.C.; Rajvanshi, A.; Chiu, H.P.; Samarasekera, S.; Rahnavard, N. C-sfda: A curriculum learning aided self-training framework for efficient source free domain adaptation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023; pp. 24120–24131.
36. Liu, Y.; Liu, J.; Shi, X.; Cheng, Q.; Huang, Y.; Lu, W. Let’s Learn Step by Step: Enhancing In-Context Learning Ability with Curriculum Learning. *arXiv* **2024**, arXiv:2402.10738.
37. van de Ven, G.M.; Soares, N.; Kudithipudi, D. Continual Learning and Catastrophic Forgetting. *arXiv* **2024**, arXiv:2403.05175.
38. Wang, D.; Zhang, X. Thchs-30: A free chinese speech corpus. *arXiv* **2015**, arXiv:1512.01882. Available online: <https://www.openslr.org/18> (accessed on 2 August 2024).
39. Surfing Tech. ST-CMDS-20170001 1 Free ST Chinese Mandarin Corpus. Available online: <https://www.openslr.org/38> (accessed on 2 August 2024).
40. Bu, H.; Du, J.; Na, X.; Wu, B.; Zheng, H. Aishell-1: An open-source mandarin speech corpus and a speech recognition baseline. In Proceedings of the 2017 20th Conference of the Oriental Chapter of the International Coordinating Committee on Speech Databases and Speech I/O Systems and Assessment (O-COCOSDA), Seoul, Republic of Korea, 1–3 November 2017; pp. 1–5. Available online: <https://huggingface.co/datasets/AISHELL/AISHELL-1> (accessed on 2 August 2024).

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.