Hindawi
Discrete Dynamics in Nature and Society
Volume 2022, Article ID 7151373, 10 pages
https://doi.org/10.1155/2022/7151373

Research Article

Federated Learning Based on OPTICS Clustering Optimization

Chenyang Lu, Su Deng, Yahui Wu, Haohao Zhou, and Wubin Ma

Science and Technology on Information Systems Engineering Laboratory, National University of Defence Technology, Chang Sha 410073, China

Correspondence should be addressed to Wubin Ma; wb_ma@nudt.edu.cn

Received 24 November 2021; Accepted 18 March 2022; Published 12 May 2022

Academic Editor: Chunrui Zhang

Federated learning (FL) has emerged for solving the problem of data fragmentation and isolation in machine learning based on privacy protection. Each client node uploads the trained model parameter information to the central server based on the local training data, and the central server aggregates the parameter information to achieve the purpose of common training. In the real environment, the distribution of data among nodes is often inconsistent. By analyzing the influence of independent identically distributed data (non-IID) on the accuracy of FL, it is shown that the accuracy of the model obtained by the traditional FL method is low. Therefore, we proposed the diversified sampling strategies to simulate the non-IID data situation and came up with the OPTICS (ordering points to identify the clustering structure)-based clustering optimization federated learning method (OCFL), which solves the problem that the learning accuracy is reduced when the data of different nodes are non-IID in FL. Experiments indicate that OCFL greatly improves the model accuracy and training speed compared with the traditional FL algorithm.

1. Introduction

With the huge improvement of algorithms and computing power in machine learning in recent years, as well as the rise of big data research, it is widely believed that artificial intelligence has ushered in the third research peak.

However, training a successful model requires a huge amount of data. With the further development of big data, it is a worldwide trend to attach importance to data privacy and security [1]. Countries are strengthening the protection of citizens' privacy security, which brings great challenges to the field of artificial intelligence. How to design a machine learning framework that allows AI systems to access the data they need without compromising data privacy, security, and regulation? One possible solution is FL [2].

FL is a collaborative training machine learning model that does not require all data to be gathered into a central server [3]; each client with data trains their own model and then synthesizes each node model to get a global model [4]. In this process, the exchange of model information between clients will be carefully designed, so that no organization can guess the private data content of another organization; this is the core idea of FL [5]. The objective of FL is to build a global model based on distributed data sets. During FL model training, model-related information can be exchanged between parties (or in encrypted form) without exposing any protected private parts of the data on each site. A trained FL model can be placed with participants or shared among multiple parties [6].

However, the traditional FL algorithm is not ideal when applied to non-IID data. Experiments show that when the data distribution of each node is highly skewed, the precision of the trained model will be greatly reduced [7]. However, the data of each node can be affected by other nodes or the local environment in the actual generation process, and the data of each node are often non-IID [8], which poses a difficult problem for the application of FL, that is, how to reduce the impact of non-IID data on the accuracy of FL.

In order to solve the abovementioned problems, this paper first uses a diversified sampling strategy to simulate data with different distributions and explores the influence of the data distribution skew degree on the accuracy of FL. The experimental results show that the training accuracy of FL model decreases with the increase of data distribution skew. In order to solve the problem of low training accuracy when the data are extremely unbalanced in FL, the local trained model parameter information of each client node is clustered by the OPTICS algorithm and divided into different clusters, so that the node distribution in the cluster has a higher similarity, then carry out training in each cluster such that each cluster gets its own global model.

In this paper, multiple nodes and parameter servers are simulated locally, and the experiment proves that it can effectively reduce the influence of the data non-IID on the model accuracy, so as to produce a more accurate model. To sum up, the main contributions of this paper are as follows.

(1) Prove that the deeper the distribution skew of each node data, the lower the precision of the global model trained in FL by experiments with real data sets;

(2) Propose a clustering method that does not need to obtain the original data of the client, the parameter server clusters the model parameters uploaded by the client through the OCFL method. OCFL has the advantages of strong applicability, insensitive parameters, and high accuracy.

(3) Simulate the non-IID distribution on multiple data sets using a diverse sampling strategy and test the effectiveness of the OCFL against the FedAvg algorithm. Experiments demonstrate that the model obtained by OCFL has higher accuracy and faster convergence.

The remainder of this paper is organized as follows. In Section 2, we provide a background on FL and an overview of related works. We then present our proposed framework, OCFL, in Section 3. Finally, in Section 4, we first show the influence of heterogeneous data distribution on the accuracy and convergence speed of the model, and then, we provide a thorough empirical evaluation of OCFL on a suite of real-world standard data sets. Our empirical results demonstrate the practical improvements of OCFL over FedAvg in heterogeneous data distribution.

2. Related Work

In order to solve the contradiction between the increasingly tighter privacy protection requirements and machine learning requirements for a large amount of training data, McMahan proposed a deep network joint learning method based on iterative model averaging and came up with the Federated Averaging (FedAvg) algorithm [9], the training of this approach takes place through a loose federation of clients coordinated by a central server; a major advantage is the separation of model training from the need for direct access to raw training data, which is significant in scenarios where data privacy is strictly required or where it is difficult to share data centrally.

With the rise of the FL study, a lot of problems emerge. Peter Kairouz et al. [10] discusses recent advances and presents an extensive collection of open problems and challenges on FL: (1) non-IID data in FL; (2) preserving the privacy of user data; (3) communication and compression; (4) robustness to attacks and failures; (5) ensuring fairness and addressing sources of bias. To improve the efficiency and effectiveness of FL, a basic challenge is the non-IID.

Non-IID exists widely in reality, for example, (1) heterogeneous client distribution, data on each client are generated locally, so the sample generation mechanism may differ from client to client (like different countries or regions); (2) characteristic distribution tilt (Covariate Shift), for example, in handwriting recognition, even if it is the same word, different people write it differently; (3) label distribution skew (Prior probability drift), such as the use of the Chinese people in China, mainly in foreign people use less; (4) quantity inclined or unbalanced, etc. In real life, all kinds of situation may lead to the occurrence of non-IID data. Traditional machine learning is based on the assumption of IID data, but FL is different from the centralized machine learning, in the case where the data are not centralized; the data for each node are non-IID [11].

In order to solve the problem of data non-IID in FL, Zhao et al. [12], improved the FedAvg algorithm, found that the FedAvg algorithm will have a high precision loss when the data are non-IID. This reduction in precision can be explained by weight divergence, which can be quantified by the distance of the Eearth mover (EMD) between the distribution of classes on each device and the population distribution, and proposed a strategy to improve training on non-IID data by creating a small subset of data that is globally shared between all edge devices. Although this method can reduce the impact of data skew, it is equivalent to artificially adding errors. Moreover, this method of data sharing essentially violates the principle of data privacy protection of federal learning and has great difficulties in implementation.

Jiang et al. [13] thought the training model can be personalized to reduce heterogeneity and get a higher quality personalized model for each model. Personalized FL can be divided into two steps: (1) build a global model in a collaborative manner and (2) personalize the global model for each client using the client's private data.

Muhammad et al. combined the FL and recommendation system and proposed the FedFast algorithm [14]. FedFast focuses on two key steps: client selection and model aggregation. In client selection, this article first uses the K-means method to cluster the similarity of recommendation systems of different nodes and divides all nodes into different classes. Then, a certain number of nodes are randomly selected from different clusters to participate in the training. Meanwhile, we used the updated gradient information of nodes that participated in training in each round to update the information of nodes of the same cluster class that did not participate in training, so as to achieve faster convergence. The main purpose of the FedFast algorithm is to improve the efficiency of training, its clustering method is to cluster the information from the recommendation system, and it is not universal to the FL method, which is not combined with the recommendation system; moreover, K-means clustering cannot exclude interference of outliers and can be attacked by malicious nodes [15]. This method requires the number of clusters to be specified in advance. In reality, the central server does not know the data distribution of the clients, so it is impossible to specify in advance how many clusters to cluster the clients.

Ghosh et al. [16] proposed the Iterative Federated Clustering Algorithm (IFCA) to divide each client into different clusters based on the local empirical loss function. The basic idea of IFCA is a strategy that alternates between estimating the cluster identities and minimizing the loss functions and thus can be seen as an alternate minimization algorithm in a distributed setting. In this paper, they prove the convergence of the algorithm for strong-convex objective function under appropriate parameter conditions and prove that exponential convergence speed can be achieved, and in a certain region, it can obtain nearly optimal statistical error rate.

Sattler et al. [17] proposed a dynamic partition algorithm based on node gradient. This paper proposes that traditional FL follows a core assumption: we can satisfy all clients with a single model. However, in fact, this is not accurate. First, the model may not be accurate enough to meet the requirements of all clients, and second, the data distribution of each client may not be the same. Therefore, in this paper, a new hypothesis is proposed: there exists a reasonable partition such that the nodes in each partition satisfy the traditional FL core assumption. This paper uses cosine similarity of each participant to divide. For a classification problem, first count the cosine similarity matrix for all the nodes, and then sort the similarity matrix by index from smallest to largest, take the smallest nodes in different groups, and merge them until finally only the group of the specified classification is left. This method also needs to specify the number of clusters in advance and cannot exclude outlier interference.

3. OCFL Architecture

In order to solve the problems mentioned above, this paper first simulates data with different distributions on multiple data sets based on different sampling strategies; experiments show that the skew degree of data distribution affects the accuracy of the model. In the case of extremely uneven distribution of data, we proposed the Clustered Federated Learning based on the OPTICS clustering method; by means of clustering, the clients in the cluster have a high similarity degree to reduce the influence of data non-IID on the model accuracy. The meanings of all symbols are shown in Table 1.

OPTICS is a density-based clustering algorithm. It defines the cluster as the maximum set of points connected by density and divides the region with sufficient density into clusters [18]. OPTICS can find clusters of arbitrary shapes in noisy spatial data compared to K-means and BIRCH, which are only suitable for clustering with convex sample sets, while OPTICS is insensitive to input parameters compared to the DBSCAN method, which improves clustering stability. To sum up, OPTICS clustering has several advantages over other clustering methods: (1) OPTICS does not require prior knowledge of the number of cluster classes to form; (2) OPTICS can find cluster classes of any shape; (3) OPTICS can detect noise points and strip out the effects of certain malicious attack nodes; and (4) OPTICS is not sensitive to input parameters. Compared to the clustering method mentioned above, the method proposed in this paper does not need to specify the number of clusters in advance and can eliminate outlier interference, so it has a wider application in practice.

In the traditional FL algorithm, a very important link is to extract a certain number of nodes from all nodes according to the turn to participate in training to improve the global model. The FedAvg algorithm adopts the method of random extraction, which randomly extracts a specified number of nodes from all nodes. This method is very effective in the face of IID data. In the face of non-IID data, the efficiency and accuracy of training will be greatly affected (see the experimental section for detailed data), and the more serious the data distribution skew, the lower the training accuracy, the non-IID data greatly affect the training quality of FL.

In the FL application scenario, the data on each node are generated independently, so the local data on each node cannot represent the overall distribution, traditional FL treats the data as IID, and it is not feasible to apply all node data with a single global model. In order to reduce the impact of the non-IID data on the model accuracy, it is a better choice to cluster users in the early stage, and then train a global model within each cluster.

θ_k is the neural network parameterization in client k, the cosine similarity between the neural network parameterization of any two clients is given by the following:

$$\begin{aligned}\alpha_{i,j} &= \alpha\left(\theta_i, \theta_j\right)\\ &= \frac{\langle\theta_i, \theta_j\rangle}{\|\theta_i\|\|\theta_j\|}.\end{aligned} \tag{1}$$

The data similarity between different clients can be obtained by calculating the cosine similarity under the same random seed of the neural network. In Figure 1, the parameters of the neural network trained by similar data are similar. Meanwhile, the parameters of the neural network trained can be regarded as high-dimensional vectors for clustering of high-dimensional vectors, which provides the possibility for clustering of nodes without obtaining node data, as long as each node uploads the model parameter information of local training.

To find the right clustering for all the client nodes, all the node clients do a full local learning and upload the learned parameters and gradient information to the parameter server, which uses the OPTICS clustering method to cluster the model parameters of these nodes into different clusters and then continue OCFL in different clusters.

To get an accurate clustering, first, all nodes receive the initial global model from the parameter server θ , and then, run SGD using local data D_k to get fully trained. After that, the new model parameters are returned to the parameter server. The neural network parameter received by the server is a high-dimensional vector composed of the parameters of each layer of neural network, which contains the local data distribution information of the client. Flatten multiple layers of data to get a $1*N$ matrix, then the OPTICS clustering is used to cluster the parameter information of each node. OPTICS clustering requires one important parameter: *min_samples* which defined

Table 1: Summary of symbols.

Symbol	Explanation
$K \leq N$	Number of clients
M	Number of data generating distribution
frac	The fraction of clients that perform computation on each round
B	The local minibatch size used for the client updates
β	The local minibatch
E	The number of local epochs
η	Learning rate
θ	Neural network parameterization
D_k	Data on client k
w_k	Model weight on clients k
$c_i \in C$	One cluster in the set of all clusters found by OCFL

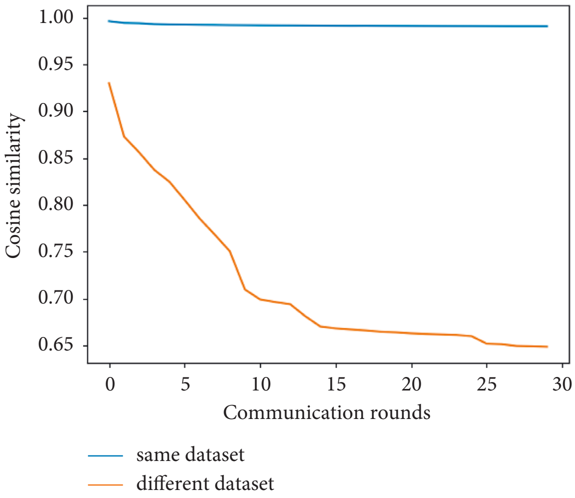


Figure 1: Cosine similarity result. The cosine similarity between model parameters from the same data set stays more or less constant throughout the FL process, the cosine similarity between model parameters from different data set quickly decreases.

the density conditions required to be a core point. Taking different values, the number of clustering results will be different; at the same time, the change of the *xi* value can also slightly change the number of clusters, the specific values of the two parameters are determined by the experimental results.

OCFL is divided into two parts: one is the operation of the parameter server and the other one is the operation of the node client. The main task of the server is to maintain the global model of each cluster; in each training round, the server randomly chooses m clients, those selected clients run SGD using local data for specified times, then sends parameter w back to server, server will sum the parameters according to the weight of the client, the more data on the client, the higher the weight. The main task of the client is to update the model parameters based on local data, receive global model parameters, and send new model parameters back.

4. Experiments

All experiments in this paper are run on a computer with the Ubantu18 64-bit operating system based on Inter(R) Core (TM) I9-9900 kF CPU @ 3.60ghz processor and GeForce RTX2080TI graphics card.

4.1. Experimental Settings. To simulate the different distribution of data, diversified sampling strategies are adopted to divide the data from the data set into different nodes according to different sampling methods. For example, when using independent sampling without replacement, randomly extract a certain amount of data from a data set, data in each node are IID (Figure 2(a)); sort the data set by label and slice it into different clients; and then data in different clients are subjected to non-IID. The skew of the data varies depending on the slice size; this paper simulates two non-IID cases: non-IID (Figure 2(b)) and non-IID2 (Figure 2(c)). In the non-IID situation, each client approximately contains two types of data; in the non-IID2 situation, each client approximately contains only one type of data, so the second is more skewed. Figure 2 shows the distribution of data when taking 10 clients.

Subsequent experiments used the Mnist and Cifar-10 standard data set, simulated the situation of 100 client nodes, data on each node account for 1% of the total data. The MLP and CNN neural networks are used for local training, to verify the algorithm under different neural network models.

4.2. Influence of Non-IID Data. Traditional FL performs differently on different distributed data. In this section, we set up a control experiment to explore the influence of the degree of skewness of the data distribution on the accuracy of the model.

Figure 3 shows the model accuracy obtained after 100 rounds of iterative training, it can be seen that with the deepening of the imbalance of data distribution, the training quality of the model also decreases. The Cifar-10 data set is a small data set used to identify universal objects, including 10 categories of RGB color images, such as aircrafts, cars, etc. Compared with the Mnist data set, Cifar-10 is a 3-channel RGB color image, and Mnist is a grayscale image. Compared with handwritten characters, Cifar-10 contains a lot of real objects, which are not only noisy, but also have different proportions and characteristics, which brings great difficulties to identification. Therefore, after 100 times iteration, the accuracy of the model on Cfar-10 data set is only about 40% (Figures 3(c) and 3(d)), which is much lower than in the Mnist data set (Figures 3(a) and 3(b)).

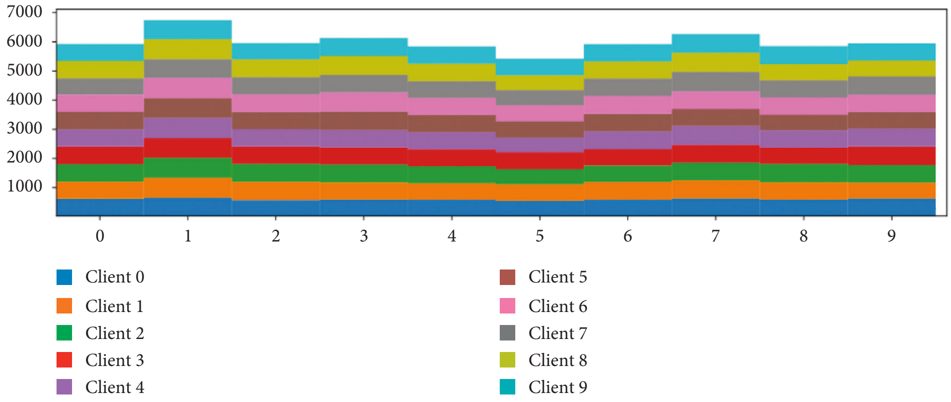
As can be seen from Figure 3, the skew of data distribution also causes the fluctuation of test accuracy; this is because when the data are extremely uneven, each client contains approximately one type of data. Then, the data extracted for each round of training will be very different, which creates an increase in volatility. In order to mitigate the effects of this fluctuation on the experiment, in subsequent experiments, relatively small learning rate and high training rounds should be selected. Specific experimental results are shown in Table 2.

```
**Input:** initial parameters $\theta$ ,$M$
**for** each $k \in M$ in parallel **do**
$\theta_k \leftarrow SG\ D(\theta, D_k)$.
$min\_samples \leftarrow 2$
$xi \leftarrow 0.2$
model ← OPTICS ($min\_samples$, $xi$)
$c_1, c_2 \ldots \in C \leftarrow$model.fit_predict$(\theta_1, \theta_2, \ldots, \theta_m)$
return $C$
```

Algorithm 1: OPTICS Clustering.

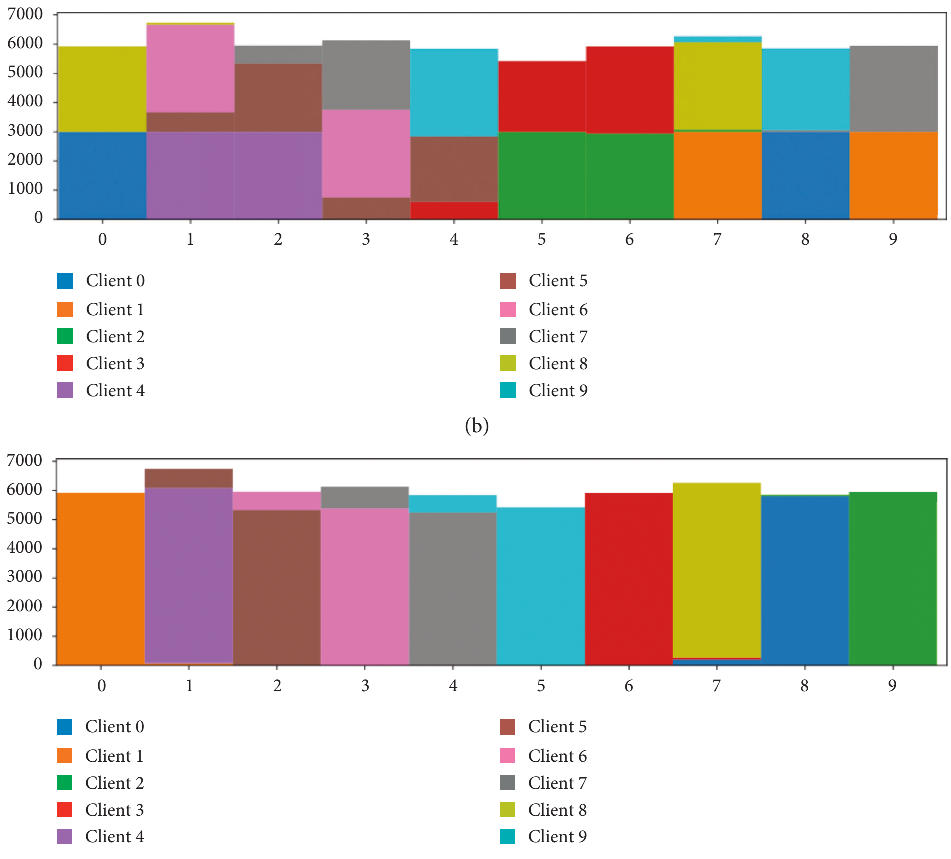
```
**Input:** initial parameters $w_0$, set of clients $c$
Server execute:
**for** each cluster $(c_1, c_2, \ldots) \in C$ in parallel **do**
**for** each round $t = 1,2,$ ...**do**
$m \longleftarrow \max(\text{frac} \cdot N, 1)$,
$S_t \longleftarrow$ (random set of $m$ clients).

**for** each client $k \in S_t$ **do**
$w_{t+1}^{k} \longleftarrow$Client execute$(k, w_t)$,
$w_{t+1} \longleftarrow \sum_{k=1}^{K} n_k / n w_{t+1}^{k}$.
return $w^{j}$ to cluster $j$ ($c_{j} \in C$)
**Client execute** ($k$, $w$)**:**
$\beta \longleftarrow$ (split $P_k$into batches of size $B$).
**for** each local epoch $i$ from 1 to $E$ **do**
**for** batch $b \in \beta$ **do**
$w \longleftarrow w - \eta \nabla \ell (w; b)$.
return $w$ to server
```

Algorithm 2: OCFL.



(a)

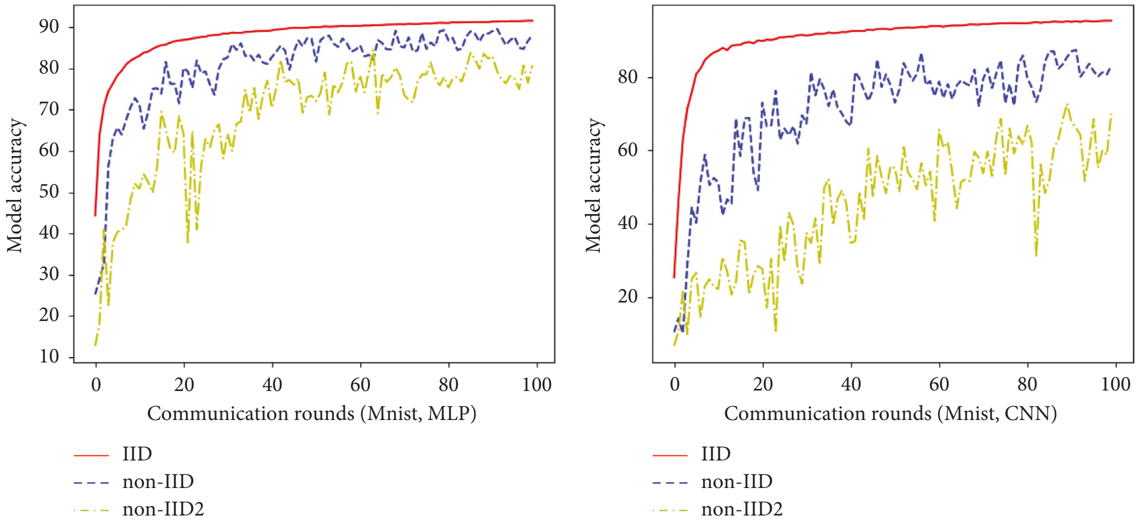
Figure 2: Continued.



(b)

(c)

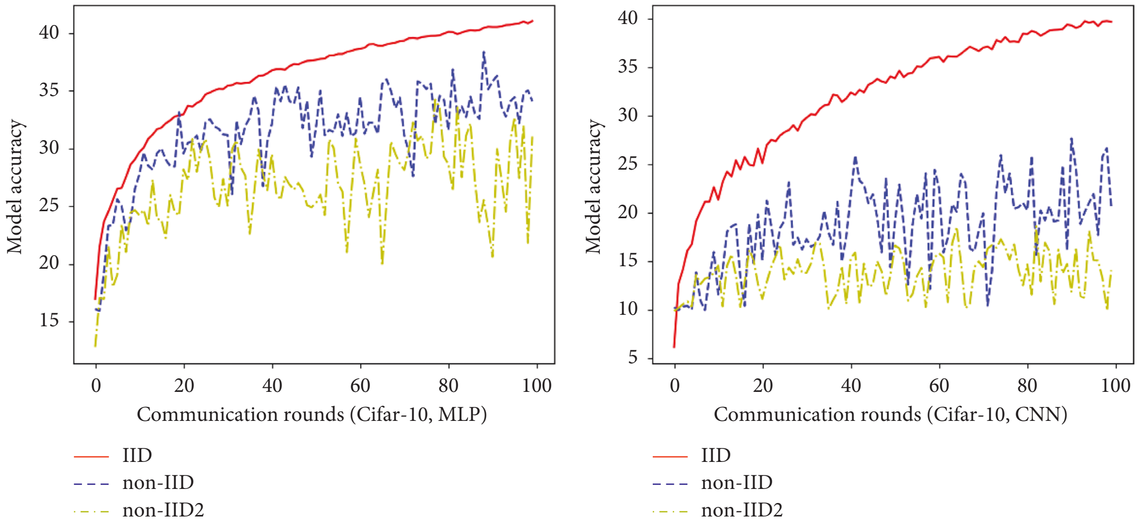
Figure 2: Display of the data distribution ((K) = 10, the first is IID, the second is non-IID, the third is non-IID2).



(a)

(b)

Figure 3: Continued.



(c)

(d)

Figure 3: Model test accuracy on different distribution data. (η = 0.001, *frac* = 0.1, (E) = 1, (B) = 10).

Table 2: Model test accuracy on different distributed data (after 100 rounds training).

	IID	Non-IID	Non-IID2	IID	Non-IID	Non-IID2
		MLP			CNN	
Mnist	0.9152	0.8797	0.8069	0.9541	0.8312	0.7006
Cifar-10	0.4103	0.3409	0.3103	0.3971	0.2070	0.1410

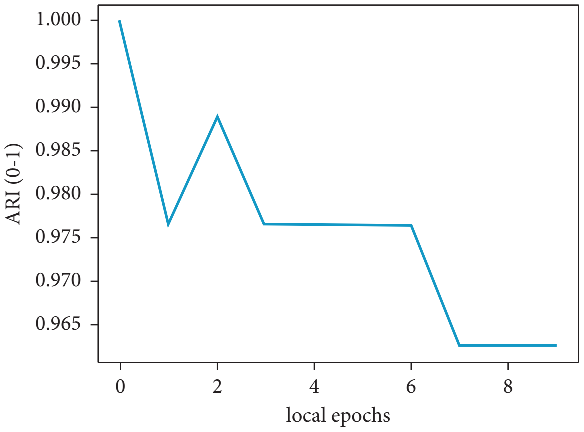


Figure 4: ARI results (data sets = Mnist, min_samples = 2, *xi* = 0.1).

Different models and data sets have different sensitivities to data distribution. In the Mnist data set, the performance of the MLP model decreased by up to 11.833%, the performance of the CNN model decreased by up to 26.570%; in the Cifar-10 data set, the performance of the MLP model decreased by up to 24.372%, the performance of the CNN model decreased by up to 64.493%. It can be seen that CNN model is sensitive to the distribution of data and is greatly affected.

4.3. OCFL Experiments. **Mnist experiments** split 60,000 instances into training (48,000) and test (12,000), the data distribution in all clients ($K = 100$) is consistent with non-IID2, and each client contains approximately one data label.

First, test the effectiveness of the OPTICS clustering, using ARI (Adjusted Rand Index) to evaluate the clustering results, the results under different local iterations epochs are shown in Figure 4.

Using OPTICS clustering can achieve relatively high accuracy and does not require a high number of local iterations. On the contrary, the effect is better when the number of iterations is low, when local epoch is 1, one can get the maximum value 1.0. Select local epochs = 1, all clients are clustered into 10 clusters and adjusted for intuitive perception of the data distribution. However, if the number of clusters is too large, the generalization of the model will be poor. While we want as many clients as possible to fit into a global model, at the same time, this clustering result is also caused by the way of data division; therefore, we decided to artificially reduce the number of clusters based on the abovementioned results. The cosine similarity between different clusters was calculated, and then similar clusters were combined; finally, 10 clusters were combined into 3 clusters.

In order to test the performance of the algorithm proposed, experiments were set to compare the accuracy of the model trained by the FedAvg algorithm and OCFL algorithm on the test set.

In Figure 5, the existence of non-IID data will cause a great fluctuation of model training; in particular, the CNN model is sensitive to data distribution, so we set a relatively low learning rate. As can be seen from the experimental results, in all clusters, the accuracy of the OCFL is higher and the convergence rate is faster (Figures 5(a) and 5(b)). Especially when the distribution of data is extremely uneven, the model trained by traditional FL may perform very poorly

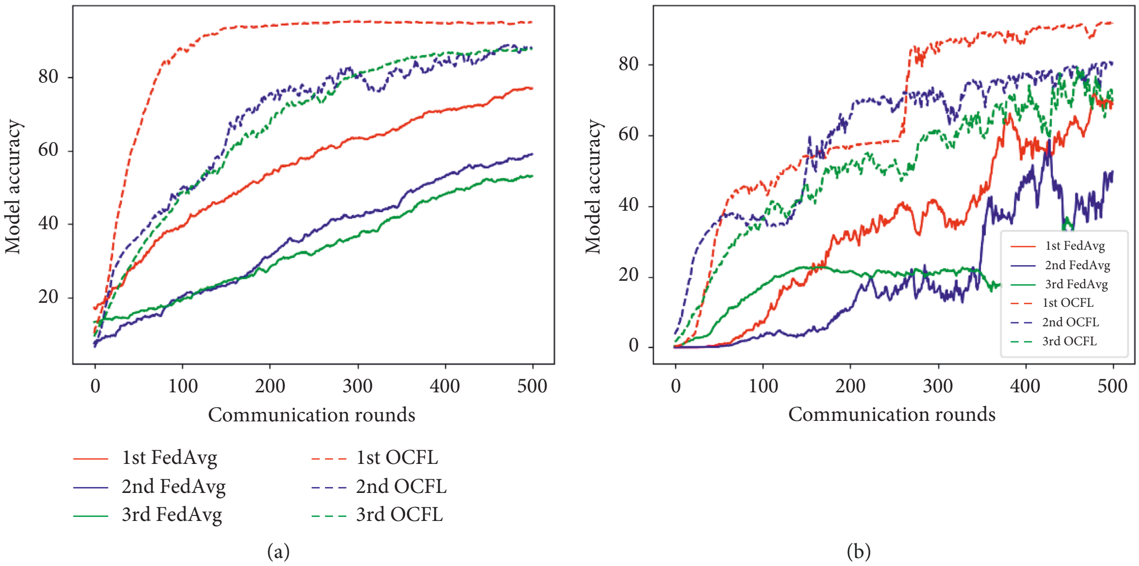


Figure 5: Mnist data set clustering experiment (rounds = 500, η = 0.000005, *frac* = 0.1, (E) = 1, (B) = 10. Left is the result of MLP, right is the result of CNN).

Table 3: Model test accuracy on Mnist.

	FedAvg	OCFL	FedAvg	OCFL
	MLP		CNN	
Cluster1	0.7806	**0.9421**	0.6895	**0.9164**
Cluster2	0.5973	**0.8823**	0.4978	**0.8081**
Cluster3	0.5321	**0.8742**	0.3029	**0.7194**

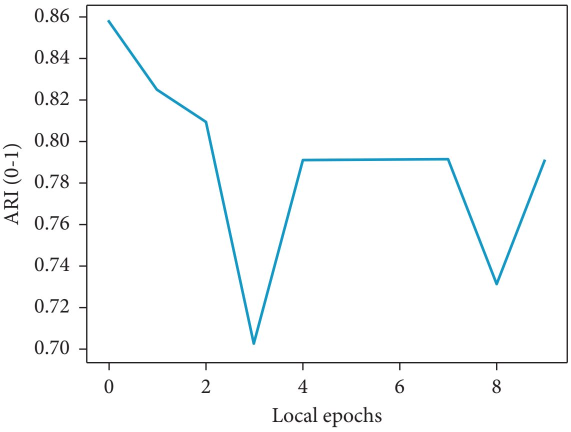
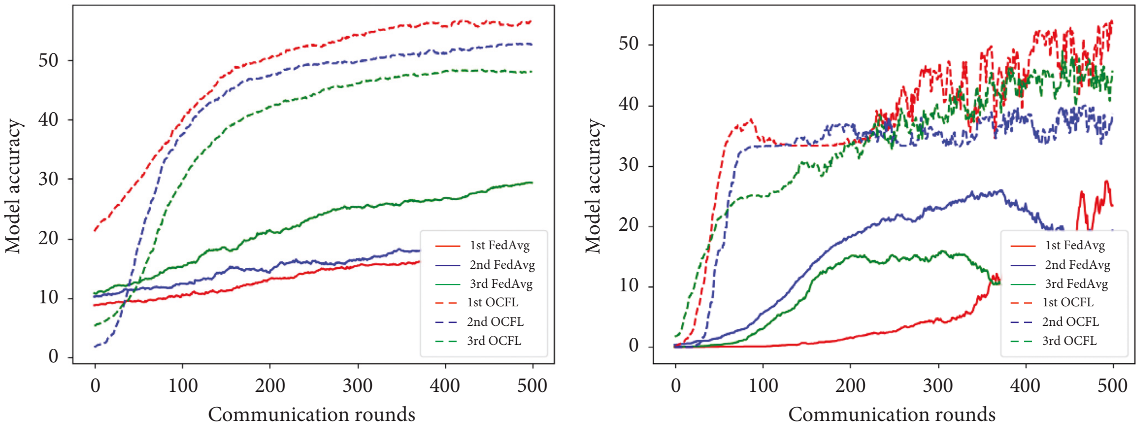


Figure 6: ARI results (Data sets = Cifar-10, *min_sample* = 2, *xi* = 0.2).

in a certain cluster; for example, as shown in Table 3, after 500 rounds training in CNN, cluster1 can get 68.95% accuracy while cluster 3 is only 30.29%. This is grossly unfair to clients in cluster 3. At OCFL, the model accuracy of each cluster training is greatly improved, the convergence speed is faster, and the imbalance of the model is alleviated.

Cifar-10 experiments split 50,000 instances into training (40,000) and test (10,000), data distribution in all clients ($K = 100$) conforms to non-IID2, where each client contains approximately one data label.

In Figure 6, the lower iteration is still selected, when the number of iterations is 1, the ARI value is 0.86, and all clients are grouped into 10 clusters. For the same reason in Mnist experiments, we calculated the cosine similarity between clusters and divided 10 clusters into 3 clusters. In Figure 7, we show the experiments in Cifar-10 data sets, the



(a)

(b)

FIGURE 7: Cifar-10 data set clustering experiment (rounds = 500, $\eta = 0.00001$, *frac* = 0.1, (E) = 1, (B) = 10. Left is the result of MLP, right is the result of CNN).

TABLE 4: Model test accuracy on Cifar-10.

	FedAvg	OCFL	FedAvg	OCFL
	MLP		CNN	
Cluster1	0.1584	**0.5642**	0.2368	**0.5413**
Cluster2	0.1903	**0.5267**	0.2065	**0.3842**
Cluster3	0.2934	**0.4797**	0.1264	**0.4516**

experimental results also demonstrate the superiority of OCFL. Specific experimental values are shown in Table 4.

5. Conclusions

This paper proposes the OCFL to reduce the impact of non-IID data on the accuracy of FL. When the local data distribution of each client is extremely heterogeneous, by clustering the model parameters of the client, the data with different distributions can be divided into different clusters according to the similarity without compromising the client's raw data. Experiments show the effectiveness of the proposed method. OCFL improves the accuracy of each cluster training model and the speed of model training; moreover, it also alleviated the fairness problem caused by non-IID to some extent.

However, there are many deficiencies in our research. We chose to comprehensively cluster all the parameters of the model, which creates huge computing and communication overhead; in future research, knowledge distillation can be used to simplify the model, reducing the consumption of calculation without reducing the clustering accuracy. At the same time, we can also use optimization algorithms such as the evolutionary algorithm to optimize the model parameters to improve the accuracy of the model.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] D. Shultz, "When your voice betrays you," *Science*, vol. 347, no. 6221, p. 494, 2015.

[2] K. Bonawitz, H. Eichner, W. Grieskamp et al., "Toward Federated Learning at Scale: System Design," 2019, https://arxiv.org/abs/1902.01046.

[3] H. B. Mcmahan, D. Ramage, K. Talwar, and Z. Li, "Learning Differentially Private Recurrent Language Models," 2017, https://arxiv.org/abs/1710.06963.

[4] Q. Yang, Y. Liu, T. Chen, and Y. Tong, "Federated machine learning," *ACM Transactions on Intelligent Systems and Technology*, vol. 10, no. 2, pp. 1–19, 2019.

[5] K. Bonawitz, I. Vladimir, K. Ben et al., "Practical Secure Aggregation for Privacy-Preserving Machine Learning," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, Dallas, Texas, USA, October 2017.

[6] Q. Yang, "AI and data privacy protection: the way to federated learning," *Journal of Information Security Research*, vol. 5, no. 11, 2019.

[7] F. Sattler, S. Wiedemann, K.-R. Muller, and W. Samek, "Robust and communication-efficient federated learning from non-i.i.d. Data," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 31, no. 9, pp. 3400–3413, 2020.

[8] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, "Federated learning: challenges, methods, and future directions," *IEEE Signal Processing Magazine*, vol. 37, no. 3, pp. 50–60, 2020.

[9] H. B. Mcmahan, E. Moore, D. Ramage, H. Seth, and A. Blaise Agüera y, "Communication-Efficient Learning of Deep Networks from Decentralized Data," 2016, https://arxiv.org/abs/1602.05629.

[10] E. Kairouz and H. B. Mcmahan, "Advances and open problems in federated learning," *Foundations and Trends® in Machine Learning*, vol. 14, no. 1, 2021.

[11] X. Li, K. Huang, W. Yang, W. Shusen, and Z. Zhihua, "On the Convergence of FedAvg on Non-IID Data," 2019, https://arxiv.org/abs/1907.02189.

[12] Y. Zhao, M. Li, L. Lai, S. Naveen, C. Damon, and C. Vikas, "Federated Learning with Non-IID Data," 2018, https://arxiv.org/abs/1806.00582.

[13] Y. Jiang, J. Konen, K. Rush, and K. Sreeram, "Improving Federated Learning Personalization via Model Agnostic Meta Learning," 2019, https://arxiv.org/abs/1909.12488.

[14] K. Muhammad, W. Qinqin, O. -M. Diarmuid et al., "FedFast: Going beyond Average for Faster Training of Federated Recommender Systems," in *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, Virtual Event, CA, USA, August 2020.

[15] A. Ghosh, J. Hong, D. Yin, and R. Kannan, "Robust Federated Learning in a Heterogeneous Environment," 2019, https://arxiv.org/abs/1906.06629.

[16] A. Ghosh, J. Chung, Y. Dong, and R. Kannan, "An efficient framework for clustered federated learning," 2006, https://arxiv.org/abs/2006.04088.

[17] F. Sattler, K. R. Muller, and W. Samek, "Clustered federated learning: model-agnostic distributed multitask optimization under privacy constraints," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 8, pp. 3710–3722, Aug 2021.

[18] M. Ankerst, M. M. Breunig, H. P. Kriegel, and J. Sander, "OPTICS: ordering points to identify the clustering structure," *SIGMOD Record: Special Interest Group on Management Data*, vol. 28, pp. 49–60, 1999.