information

Article

FedUA: An Uncertainty-Aware Distillation-Based Federated Learning Scheme for Image Classification

Shao-Ming Lee [1] and Ja-Ling Wu [1,2,*]

[1] Department of Computer Science and Information Engineering, National Taiwan University, Taipei 10617, Taiwan; gene840802@gamil.com
[2] Graduate Institute of Networking and Multimedia, National Taiwan University, Taipei 10617, Taiwan
* Correspondence: wjl@cmlab.csie.ntu.edu.tw

Abstract: Recently, federated learning (FL) has gradually become an important research topic in machine learning and information theory. FL emphasizes that clients jointly engage in solving learning tasks. In addition to data security issues, fundamental challenges in this type of learning include the imbalance and non-IID among clients' data and the unreliable connections between devices due to limited communication bandwidths. The above issues are intractable to FL. This study starts from the uncertainty analysis of deep neural networks (DNNs) to evaluate the effectiveness of FL, and proposes a new architecture for model aggregation. Our scheme improves FL's performance by applying knowledge distillation and the DNN's uncertainty quantification methods. A series of experiments on the image classification task confirms that our proposed model aggregation scheme can effectively solve the problem of non-IID data, especially when affordable transmission costs are limited.

Keywords: federated learning; model aggregation; knowledge distillation; uncertainty in deep neural networks

Citation: Lee, S.-M.; Wu, J.-L. FedUA: An Uncertainty-Aware Distillation-Based Federated Learning Scheme for Image Classification. *Information* **2023**, *14*, 234. https://doi.org/10.3390/info14040234

Academic Editors: Amar Ramdane-Cherif, Ravi Tomar and TP Singh

Received: 27 February 2023
Revised: 4 April 2023
Accepted: 6 April 2023
Published: 10 April 2023

1. Introduction

The concept of FL was proposed by McMahan et al. in 2016 [1]. Its goal is to complete the training of a global model when target datasets are distributed to different devices (or clients), and the sensitivity of each dataset is of grave concern. In addition, the authors [1] also proposed a federated averaging (FedAvg) algorithm to complete the task of global aggregation, so that each client can complete the model training and keep their data locally. FedAvg prevents users from transmitting sensitive data from their side to the server by uploading the client-side's model gradients to the server instead. Then, the server aggregates the uploaded gradients to build a new global model to protect the privacy and security of every client's local data.

Although FedAvg claims to be able to deal with non-IID data, many studies have pointed out that the accuracy of FedAvg seriously drops if the processed data are non-IID [2,3]. The main reason for the performance degradation is that the non-IID data will cause the weights of the local models to diverge. More precisely, since the loss function of a regular neural network (NN) is non-convex, if FedAvg obtains the global model by conducting the mean operations, it will continuously increase the gap between the obtained result and the ideal model obtained by training on ideal IID datasets, which in turn makes the ensemble unable to converge, and deteriorates the learning performances [4]. In addition, FedAvg cannot fully utilize all of the information provided by clients, such as the inter-client gradient variations.

Currently, the primary methods for dealing with the problem of non-IID data can be divided into three categories: data-based, system-based, and algorithm-based [5]. The data-based category solves the non-IID problem directly and effectively through data sharing [3,6] or data augmentation [7] techniques. However, such methods often violate

the spirit of FL because there is a risk of data privacy leakage due to the inability to practice data decentralization securely. In contrast, system-based methods usually use clustering techniques to cluster users for the construction of multi-centric frameworks [8,9], and users in the same group will have similar training data. The adopted data similarity estimation methods can be further divided into two types: estimating the similarity of the loss values and estimating the similarity of the user-end model weights. The realization of algorithm-based methods are very diverse and include regularization [10,11], fine-tuning [12], and personalization layers [13], and these are introduced in user-end training. There are also some standard techniques in machine learning, such as multi-task learning [14], lifelong learning [15], and knowledge distillation [16–20].

Guha et al. proposed DOSFL [16] as a "one-shot" FL architecture. Unlike the model distillation method, this architecture uses the dataset distillation method, in which the client distills the local data and uploads the synthetic data and learning rate to the server. The server combines the synthetic data from the users to train a global model. Jeong et al. proposed the federated distillation (FD) architecture [17], in which users upload the per-label mean logit vectors for each label to speed up communications. In addition, for facing non-IID problems, a federated augmentation (FAug) [17] algorithm is proposed to deal with them. FAug will ask all users to inform the server of the samples they lack, the algorithm will let the server train a GAN, and then then allow the user to download the GAN to expand their local data into IID patterns. Compared with FedAvg, FD, and DOSFL, FAug can significantly reduce communication costs, but the associated accuracy performance is somewhat poor. The architecture of FedMD [18], proposed by Li et al., requires a public dataset. The user first uses the public dataset for general training and the local private data for customized training. In the communication stage, the user uploads the logarithmic probability calculated from the public dataset, and the server averages the logarithmic probability uploaded by all users before learning. Compared with FedMD, FedDF [19] proposed by Lin et al. uses unlabeled data for distillation and transfers the distillation task from the user to the server side. The results show that FedDF has a better robustness in selecting distillation datasets and is suitable for the context of FL. Figure 1 shows the block diagram and information flow of FedDF, which is chosen as the major benchmark for our newly proposed work. Each of the indicated functional modules of Figure 1 will be detailed in the next Section. Chen et al. proposed FedBE [20]. The architecture of FedBE is based on FedDF, and it introduces the Bayesian inference for sampling more models, and applies Bayesian ensembles to obtain better global models. FedBE has been proven effective for resolving non-IID problems, and is compatible with other architectures that normalize user-side models.

As mentioned above, our work was developed based on the architecture of FedDF, but with the following notable characteristics:

1. The server quantifies the network's uncertainty of the uploading client, which serves as the basis for building a more adaptable aggregation scheme to deal with the inhomogeneity of client side models;
2. The server introduces the sample's quality evaluation to effectively sieve through samples to suppress the influences of data uncertainty and improve learning efficiency;
3. As a knowledge distillation aggregation architecture, our work can effectively separate the information of uncertainty and inter-class relationships. This separation helps solve the non-IID data issue and provides a good learning performance while limiting the transmission costs.

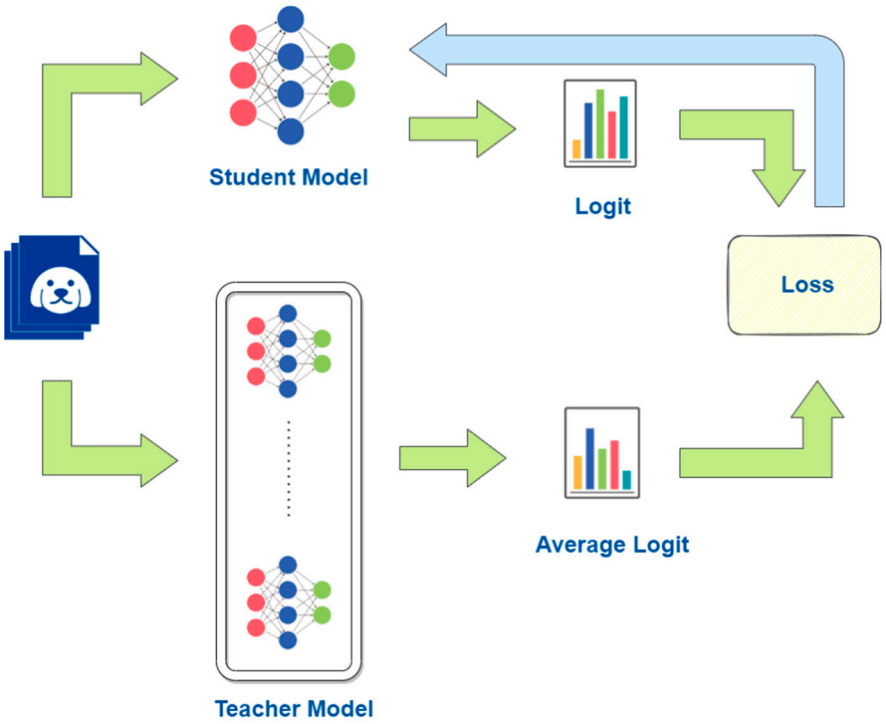


Figure 1. The schematic diagram and information flow of FedDF [19] (The green-color arrows indicates the forward training direction and the blue arrow the backward training one).

2. Preliminary Backgrounds

2.1. Knowledge Distillation

Initially, knowledge distillation was proposed to be applied for model compression [21], where the goal was to compress one or more large models (teacher models) into small models (student models). The resultant small models could effectively learn the so-called "important knowledge" from the pre-trained large models, allowing them to enhance a certain level of effectiveness associated with a specific requirement. Knowledge distillation is generally used to make small models have a better generalization ability. For example, as shown in Figure 2, a knowledge distillation-based classifier can effectively learn inter-class relations (a.k.a. dark knowledge) by regulating the distillation temperature in classification problems.

Knowledge distillation is a promising idea for federated learning. There are two practical reasons to support the above claims: First, it can alleviate over-fitting on the user side: In the context of FL, if users cannot perform model aggregation frequently due to the communication limitation, the differences in models among users will continue to accumulate, and the user model will learn too many useless local data features, compared to the general global IID data assumption in the server, those useless features behave like redundant noises, which in turn handicap the final aggregation result to approximate that of the ideal model. Applying knowledge distillation in the aggregation stage helps the global model sift through informative and valuable information for learning; therefore, it can alleviate the bias caused by incomplete or overtrained data that often occurs on the user side in FL.

As addressed above, knowledge distillation enables the global model to learn the inter-class relationship, which helps transfer the knowledge learned for a general multi-purposed model to a specific target-oriented model; this is the second reason for using knowledge distillation in FL. To dive into the reasoning of this claim in more detail: when the data are not independently and identically distributed, the inter-class relationship learned by the local model may be incomplete and inconsistent. An inappropriate aggregation scheme may not effectively transfer the genuine inter-class relationship to the global model.

Of course, effectively identifying the inter-class relationships suitable for FL becomes a research topic worthy of deeper investigation.

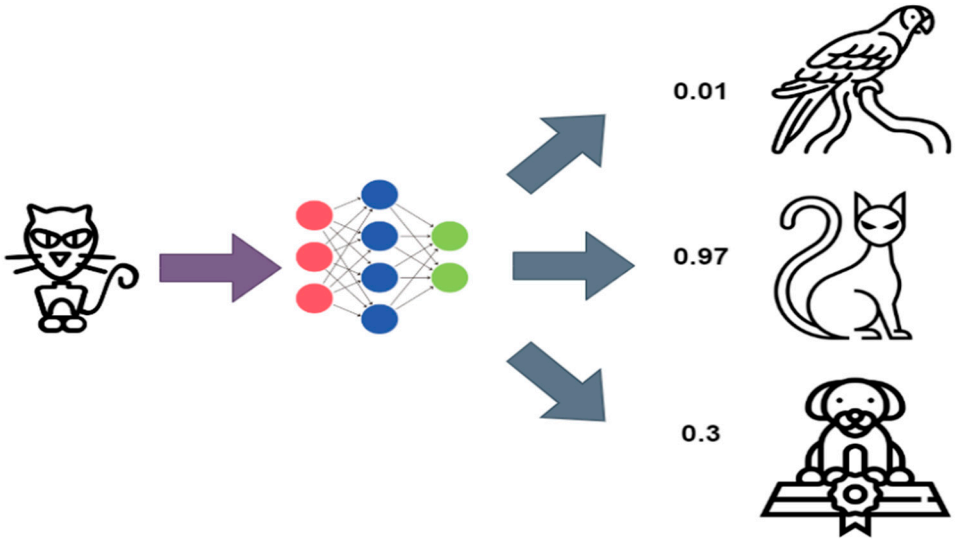


Figure 2. An illustrative example of learning dark knowledge through a knowledgeable teacher model in image classification problems. In this example, although we know that the input image falls in the cat category, we also know that the input has a higher probability of falling into the dog category than into the bird category. The pre-described classification order is the so-called "inter-class relationship".

However, the current distillation-based federated learning architectures have yet to effectively consider all of the advantages mentioned above. The uncertainty-aware distillation-based federated learning (shortened as FedUA) scheme proposed in this paper aims to provide a possible solution to improve the learning effect when both the non-IID data and the limited communication capacity occur at the same time.

2.2. Uncertainties in DNNs

General DNNs cannot express confidence levels; however, displaying confidence levels is increasingly essential for specific application domains, such as safety-critical tasks and medical applications. Therefore, studies on the uncertainty of NNs have also been investigated, including defining the sources of uncertainties in DNNs, quantifying uncertainties through various measures, and constructing correction networks, to name a few.

Generally speaking, the uncertainties of NNs can be divided into the following three types:

1. Data uncertainty—the uncertainty inherent in the data; even with a well-calibrated model, such an uncertainty still exists;
2. Model uncertainty—the model needs to be built with more knowledge. Generally speaking, this kind of uncertainty can be suppressed by improving the training process or calibrating the model;
3. Distributional uncertainty—the uncertainty of the distribution prediction itself. From another viewpoint, such an uncertainty can be an essential basis for out-of-distribution detection [22–25]. Figure 3 shows the classification of the uncertainties of NNs. We refer interested readers to find the detailed definition of each uncertainty class in [22].

Data uncertainty will manifest in the final forecast, such as estimating the outputs of a normalized exponential function (a.k.a. the Softmax output) for a classification task or the standard deviation of the predictions for a regression task. However, studies have found that NNs often suffer from overconfidence, and the normalized exponential function output is often poorly calibrated [26–28], resulting in imprecise uncertain estimates.

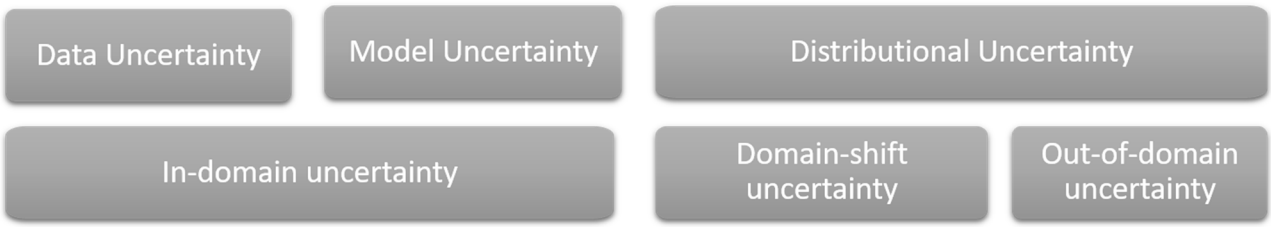


Figure 3. The Classification of the uncertainties of neural networks.

For FL, the adverse effects are more pronounced and trickier when the client data causes distributional uncertainty. Therefore, quantifying uncertainty is undoubtedly a critical information basis for model aggregation. This work investigates an FL architecture based on knowledge distillation (cf. Figure 4). We use the logarithmic probability extracted by the teacher model as the primary basis (i.e., the confidence measure) for the training process of the student model. If the confidence of the current sample can be effectively calculated under the multi-teacher structure, a more flexible and efficient knowledge transfer can be made successfully. Our proposed FedUA considers the influences of the uncertainties mentioned above and adds uncertainty quantification steps to clarify the global model's training objectives in the aggregation stage.

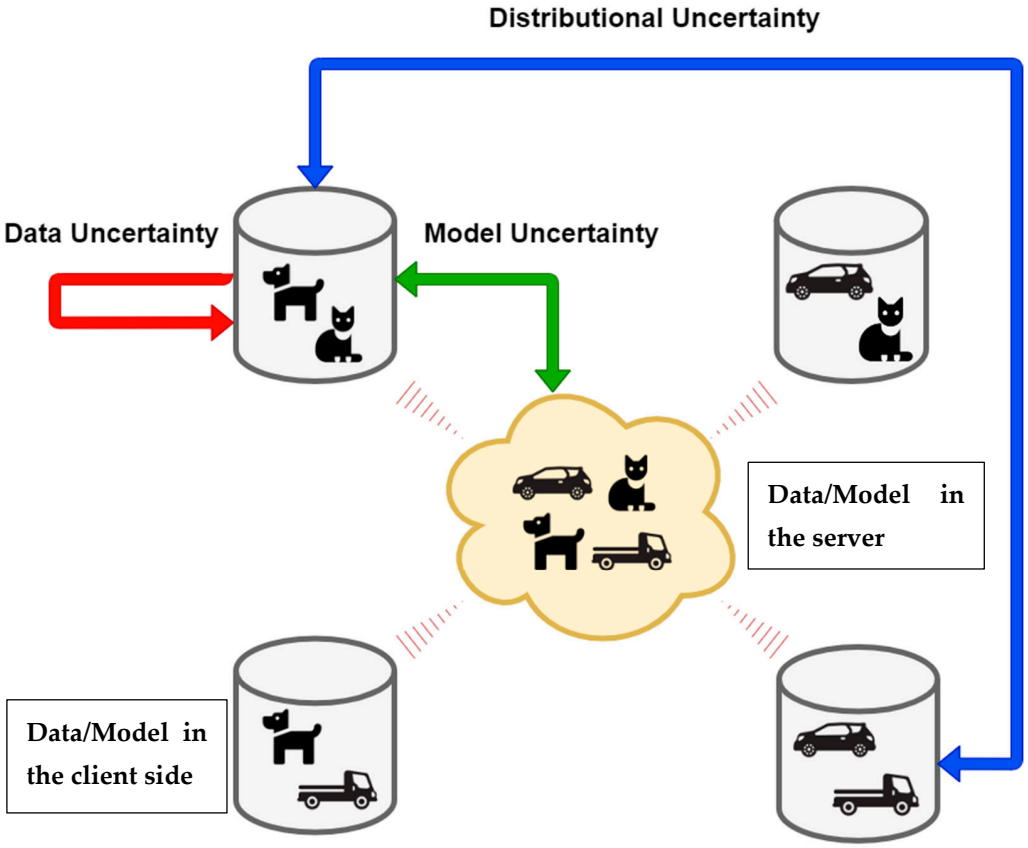


Figure 4. NN Uncertainties under the structure of federated learning.

3. The Proposed Method: FedUA

We designed our FL framework based on the so-called knowledge distillation-aware aggregation scheme to conquer the challenges of non-IID client data and the restricted communication between clients and the server. We add two core functional modules: the uncertainty measurement and the sample quality evaluator to enhance the overall system performance. The following subsections will describe these two modules' operations and architectures in detail.

3.1. Uncertainty Measurement

Due to the settings of FL, each client uses local data as the training dataset of the NN, resulting in a regional model; even under the same training parameters, the same data input may still produce highly inconsistent losses and predictions during the testing phase. The server side is decentralized in data, making it impossible to spy on or dig out which kind of data prediction the user model is good at. We bring the uncertainty measurement into the DNN as a vital basis for conducting the model aggregation process so that the server side can catch the confidence level of each participating user in the prediction generated by the specific input data to generate the subsequent integration results and strengthen the reliability of the global model.

Considering the knowledge distillation-based FL architectures, it is expected that in the aggregation stage, one can use referential information to approach the outcome of an ideal teacher model to perfect the knowledge inheritance. According to the model uncertainty, if an enormous amount of input data belongs to a specific object category in the model learning stage, the trained model should produce higher confidence concerning the output of this category in the inference stage. Regarding the distribution uncertainty, through practical measurements, all of the teacher models can participate in teaching the student models by "making use of their strengths and circumventing their weaknesses," which further enables the server side student models to have a more comprehensive classification ability.

To accommodate the variations of each client's data, we use the Gaussian discriminant analysis for each client to establish a Gaussian mixture model of its characteristic spatial density. Given a set of (X, Y), the establishment method is as follows:

for each class c with samples $X_c \subset X$ **do**

$$w_c \leftarrow \frac{|X_c|}{|X|} \tag{1a}$$

$$\mu_c \leftarrow \frac{1}{|X_c|} \sum_{X_c} f_w(X_c) \tag{1b}$$

$$\sigma_c \leftarrow \frac{1}{|X_c|} \left(f_w(X_c) - \mu_c\right) \left(f_w(X_c) - \mu_c\right)^T \tag{1c}$$

Prior to the model aggregation, a Gaussian mixture model is used to quantify the epistemic uncertainty of the current sample for a specific user-end model. The process is as follows:

$$z \leftarrow f_w(x) \qquad (f : a\ feature\ extractor) \tag{2a}$$

$$p(z) \leftarrow \sum_{\mathrm{c}} w_c\, N(z; \mu_c; \sigma_c) \qquad (N : Gaussian\ model) \tag{2b}$$

For a given user-end model, we input sample x into feature extraction function f to obtain feature vector Z and its corresponding p(z). The feature space density probability of the server side samples associated with the current client side model can now be calculated.

At this time, the uncertainty measurement method we adopted is called the single deterministic model, that aims to reduce the computational burden of the model during training and testing. In addition, we used feature space as the quantization objective instead of the normalized exponential function (i.e., the SoftMax). The reason for this is because under the knowledge distillation-based FL architecture, the inter-class correlation of the data is beneficial to the aggregation model, and this relationship is reflected in the aleatoric uncertainty. Therefore, the aggregation process can exclude the influence of this factor to avoid the occurrence of an objective mismatch. Because of this consideration, we also made a comparative analysis in our experiment.

3.2. Sample Assessment

For typical knowledge distillation, the training data of the student and the teacher models are independently and identically distributed so that the two can achieve an efficient and stable knowledge inheritance. However, considering the situation of many teachers under the structure of FL, the teacher model uploaded by a client is prone to overfitting the local data. We hope that the server aggregation stage can effectively bring the global model towards a more generalized direction to eliminate this shortage.

To achieve the above purpose, we should carefully select the students' training data, so we include a sample evaluator in FedUA (cf. Figure 5) to be responsible for the sample evaluation task. At this stage, we followed the spirit of active learning and select samples with high epistemic uncertainty as the training data for the teacher model. We adopted the Bayesian active learning by disagreement (BALD) technique [29], that quantifies the uncertainty of the samples based on the Bayesian viewpoint, and mathematically it can be written as:

$$I(y; w|x, D) = H(y|x, D) - E_{p(w|D)}[H(y|x, w, D)], \tag{3}$$

where H (x | y) denotes the conditional entropy of x given y, and I (x, y | z) represents the conditional mutual information between x and y given z, respectively.

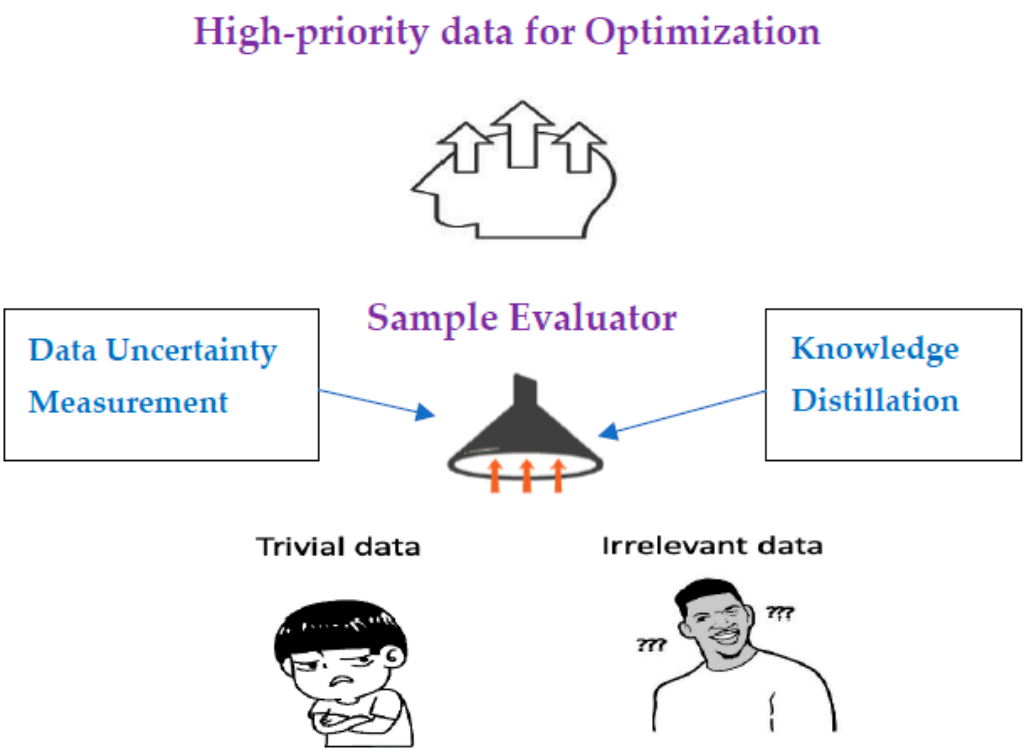


Figure 5. The effects of the sample evaluator (with the aid of knowledge distillation and measurement of data uncertainty).

In other words, we aim to find samples x, that maximize the mutual information between model output y and the model parameters {x, D}. From an information-theoretical point of view, the qualified samples should meet the following conditions: (1) Low confidence in average model output and (2) high confidence in a single sampling model output. Based on the above, for the samples with more prominent mutual information, it is harder to achieve consensus on the outputs between the models; therefore, they are what we are seeking.

In practice, the well-known Monte Carlo approximation can simplify the computation of the conditional mutual information in the above equation. That is,

$$I(y; w|x, D) \approx -\sum_c (\frac{1}{T}\sum_t p_c^t) \log\left(\frac{1}{T}\sum_t p_c^t\right) + \frac{1}{T}\sum_{c,t} p_c^t \log(p_c^t), \quad (4)$$

where p_c^t denotes the output probability of class C for model T.

When applied to FL, as pre-described, we can comprehend the distillation process (cf. Equation (3)) as "samples that do not reach consensus among local models", should be taken with higher priority.

Samples with this characteristic will have a considerable divergence in the direction of model convergence during the training phase. Hence, they are more important for optimizing the global model on the server side. In our realization, the samples generated values computed from Equation (4) that are higher than a given threshold will be denoted as high-priority samples for optimization. Of course, the threshold value is accuracy-sensitive and is application dependent. In our experiments, this is empirically determined during the simulation iteration.

3.3. Overall Architecture

Under the mechanism of knowledge distillation, we hope that the student model can learn the inter-class relation of the ideal model well to suppress the adverse effects of data uncertainty. However, if the adopted uncertainty measurement is highly susceptible to data inhomogeneity, it will also be a disadvantage for the proposed FedUA. For example, suppose there is a sample with high data uncertainty from the viewpoint of the ideal model. For such a sample, the associated uncertainty measurement will output a low confidence. In contrast, from the perspective of class distinguishability, the more representative client (who can demonstrate a better interclass relation) will show a decrease in confidence value for this sample due to its native data uncertainty. This fact will degrade the overall performance of our FedUA. We found this problem when we tried to add the uncertainty measurement to the knowledge distillation-based FL, where the entropy of Softmax outputs of the NNs is applied to measure the data uncertainty. This finding explains why in our realization, we replace the Softmax entropy with its feature space density's counterpart (cf. Equation (2a)). Moreover, our experimental results, as illustrated in the next Section, will also justify that feature space density is less affected by the samples' native data uncertainty than that of the Softmax outputs of the NNs.

Figure 6 shows the schematic diagram of the overall architecture of our proposed FedUA. FedUA comprises two main boxes: the server box and the client box. As shown in the upper portion of Figure 6, the server box consists of five functional modules: the teacher evaluation, the sample assessment, the uncertainty measurement, the logits computation, and the student learning modules (note that the brown-colored arrows indicate the respective information flows of each functional module).

In each round, the server regards all user-end models uploaded in this round as the teacher model. When performing the teacher model evaluation, we capture the forward pass outputs of a specific NN layer and send them to the sample assessment and the uncertainty measurement modules for further analyses. The uncertainty measurement uses the selected features of the user-end model to represent the model outputs' weights. Instead of the original FedDF averaging operation in the logits combination module, we apply those weights to calculate the combined logarithmic probabilities (a.k.a. the ensemble logits). At the same time, the user-end model's prediction values are used for the sample quality evaluation, and the qualified samples (with prediction values more prominent than a pre-defined threshold) will be chosen as the training data of the teacher model. Then, after performing these preprocesses, the average parameters of the teacher models (computed through FedAvg) will be treated as the initial parameters of the student model. Then, we can perform the subsequent knowledge distillation (as indicated in the student learning module of Figure 6, we use the Kl-divergence to complete the corresponding calculation).

Following the end of the knowledge distillation, the trained student model will be sent back to the users as the global model for conducting the following local training (as indicated in the client box at the bottom of Figure 6).

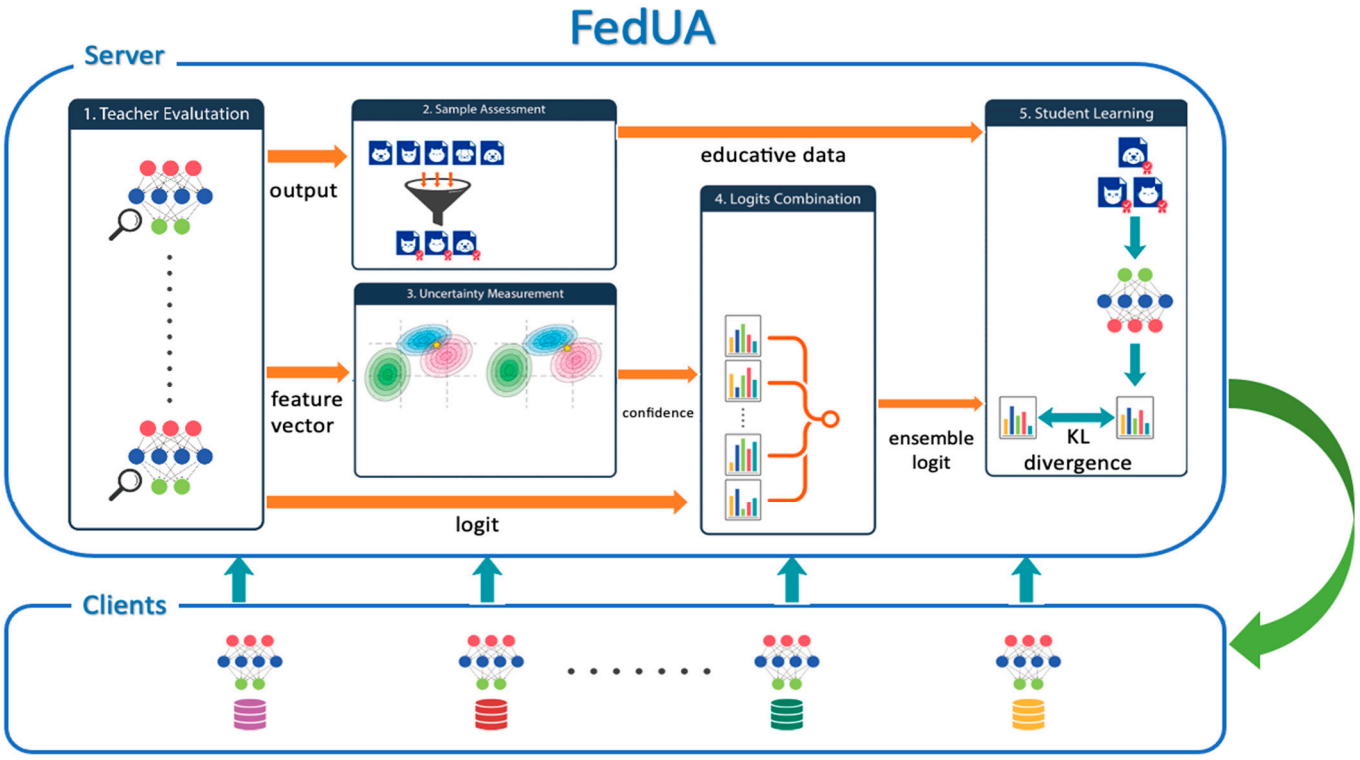


Figure 6. The schematic diagram and the information flow of our proposed FedUA.

4. Experiments

To verify the effectiveness of the adopted core methods, we conduct ablation analyses on the uncertainty measurement and sample evaluation, respectively. All experiments are repeated more than five times. The statistical average and variances will be reported as our experimental results.

4.1. Experimental Settings

(a) **Datasets and Network Models**

We examined the proposed FedUA architecture in an image classification application. We selected ResNet-32 as the benchmarking neural network architecture and CIFAR-10 as the training dataset. We randomly picked 40,000 images from the training data as label data for local training on the client side. The remaining 10,000 images were used as unlabeled data for the server side distillation aggregation.

For the label data used for client training, we used the step method [20] as the baseline to achieve the goal of non-IID, and the Dirichlet to make different types of non-IID patterns. Under the step method, each client had many images of two specific categories and a few pictures of the remaining eight categories. The Dirichlet method uses a concentration parameter α (a.k.a. the concentration parameter), to regulate the Dirichlet distribution to produce data with different degrees of dispersion.

CIFAR-10 comprised ten categories of data composed of various vehicles and animals. The existence of inter-category relationships is beneficial for us to explore the correlations between the knowledge distillation, the uncertainty measurement, and the federated learning architecture. The obtained correlations helped to confirm the ability to learn the relationship between the classes and judge the effectiveness of the teacher model in the aggregation stage of federated learning.

(b) **Detailed Processes**

In order to facilitate comparisons and consider the parameter settings concerning related works, we set 40 rounds as the upper limit. The number of clients was assumed to be 10, and the reporting fraction was initialized to 1.0. The reporting fraction determines the number of randomly selected customers in each round, representing the proportion of models uploaded for subsequent aggregation.

Each round of local or server side training consisted of 20 epochs and applied the commonly adopted stochastic gradient descent method. We set the batch size to 32 on the local side and 128 on the server side. In addition, to adapt to knowledge distillation, relative entropy (i.e., the KL divergence) was used on the server side, and this is different from using cross entropy as the loss function on the local side.

4.2. Results and Analyses

4.2.1. Ablation Analysis

(a) **The Impact of the Sample Assessment**

In the ablation analysis of the sample assessment, we verified the effectiveness of Bayesian active learning by disagreement (BALD) first. Then, we considered the impact of the different sample ratios (SRs) on the unlabeled data.

We used random batch sampling as the benchmarking target for a fair comparison. Table 1 presents the relevant results of this examination, where we only depict the portions with a fixed unlabeled data sampling ratio because our experimental results demonstrate that BALD performs better under the condition associated with the same unlabeled data sampling ratio. Moreover, the results listed in Table 1 confirm that using BALD to screen out sample batches demonstrates a better meaning in learning, and therefore performance in accuracy, for the global model's optimization than using random batches traditionally.

Table 1. The performances under different settings in the sample assessment test. (Benchmark NN architecture: ResNet-32, training dataset: CIFAR-10, SR: sample ratio).

	SR = 0.2	**SR = 0.4**	**SR = 0.6**	**SR = 0.8**	**SR = 1.0**
Random	71.5 ± 0.61	71.3 ± 0.78	71.8 ± 0.66	**72.3 ± 0.66**	72.1 ± 0.55
BALD	72.0 ± 0.24	72.5 ± 0.36	**73.9 ± 0.46**	73.7 ± 0.33	73.2 ± 0.48

Interestingly, we also found that regardless of which filtering method was adopted, the best performance in some cases (other than iterative training) occurred with a complete dataset. For example, the best-performed SR setting for random and BLAD filtering is 0.8 and 0.6, respectively (cf. the boldfaced items in Table 1). A smaller sample ratio stands for less induced computational loads.

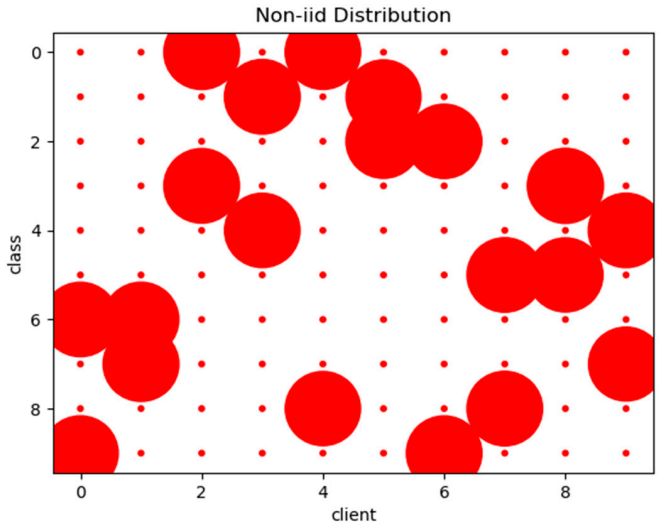
In conclusion, adding our proposed sample evaluation mechanism is beneficial, not only for the performance of distillation federated learning, but also helpful in reducing the computational burden on the server side.

(b) **The Impact of the Uncertainty Measurement**

In the rest of this subsection, we focus on the effectiveness of the uncertainty measurement. We calculate and compare the entropy of the feature space density outputs and the Softmax outputs in the inference mode when clients learn with non-IID data under the original learning settings.

CIFAR-10 has a high degree of data uncertainty because there is a specific correlation among the animal classes, and the same is valid for the vehicle data. Due to the assumptions of non-independence, we should pay attention to both model effects and distribution uncertainties. The former is because the data imbalance at the category level will affect the local model. The latter comes from the distribution uncertainty when the uploaded local model is compared with the ideal global model, that unavoidably has a distribution difference between the training and actual samples.

We take the example of a non-IID in Figure 7 for illustration. Client numbers 0, 2, 3, 5, 6, and 7 contain many images in two categories: vehicles and animals, so these clients should behave in a superior manner in the coarse-grained classification task. In contrast, clients numbers 1, 8, and 9 contain many images in two animal classes, and client number 4 contains many images in two vehicle classes. These clients tend to have specific behaviors in fine-grained classification associated with their highly correlated classes.



Class Number	0	1	2	3	4	5	6	7	8	9
Images	Airplane	Automobile	Bird	cat	Deer	Dog	Frog	Horse	Ship	Truck

Figure 7. An example of non-IID training data distribution (horizontal-axis: client number and vertical-axis: image class number).

For example, when the input sample belongs to the animal category, the No. 4 teacher model incorrectly classifies coarse-grained and fine-grained animal categories. Therefore, we should suppress the degree of the student model's referencing to the No. 4 teacher model. In addition, promoting the fine-grained ability of teacher models 1, 8, and 9 for animal classes is crucial in improving the student model's training effects for the later stages. Conversely, if the input sample belongs to the transportation category, we should lower the influences of No. 1, 8, and 9 teacher models. At the same time, the impact of the No. 4 teacher model should be increased in the aggregation stage.

Figures 8 and 9 show the distributions of the top-1 outputs' entropy of Softmax and feature space density, respectively. For ease of comprehension, we respectively illustrate the mean entropy values of Figures 8 and 9 in Figures 10 and 11. To emphasize the different behaviors of the two distributions in real applications, say out-of-distribution (OoD) detection as an example, let us explore the two-dimensional distribution patterns of the two in detail, as indicated in the red-colored rectangular boxes in Figure 12. Clearly, from Figure 12, the latter is a better choice than the former due to its higher sparsity in distribution.

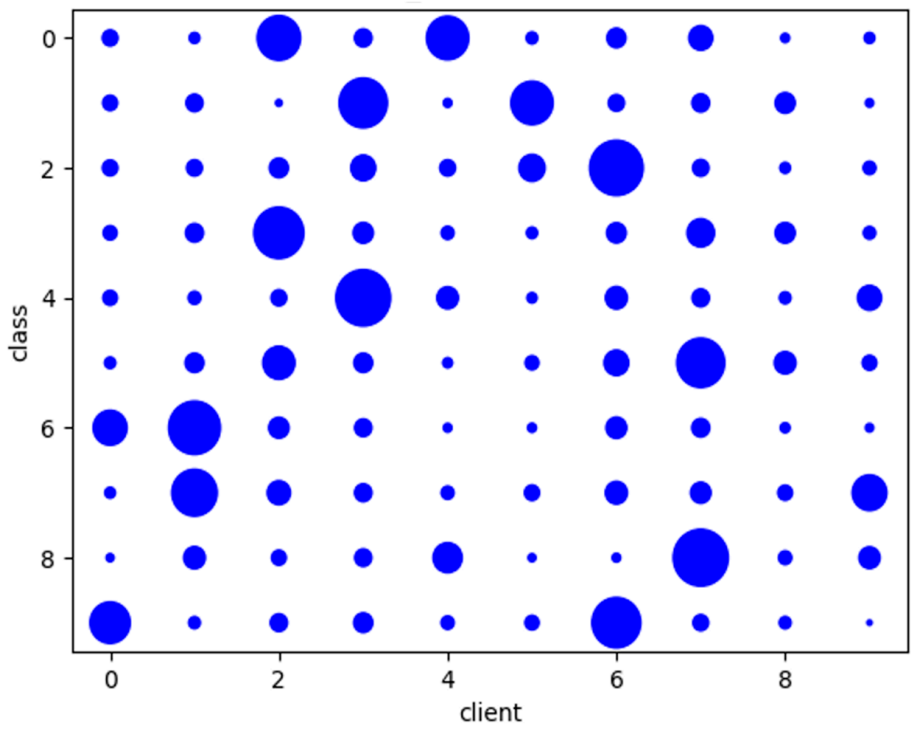


Figure 8. The statistical distribution of the top-1 SoftMax outputs' entropy.

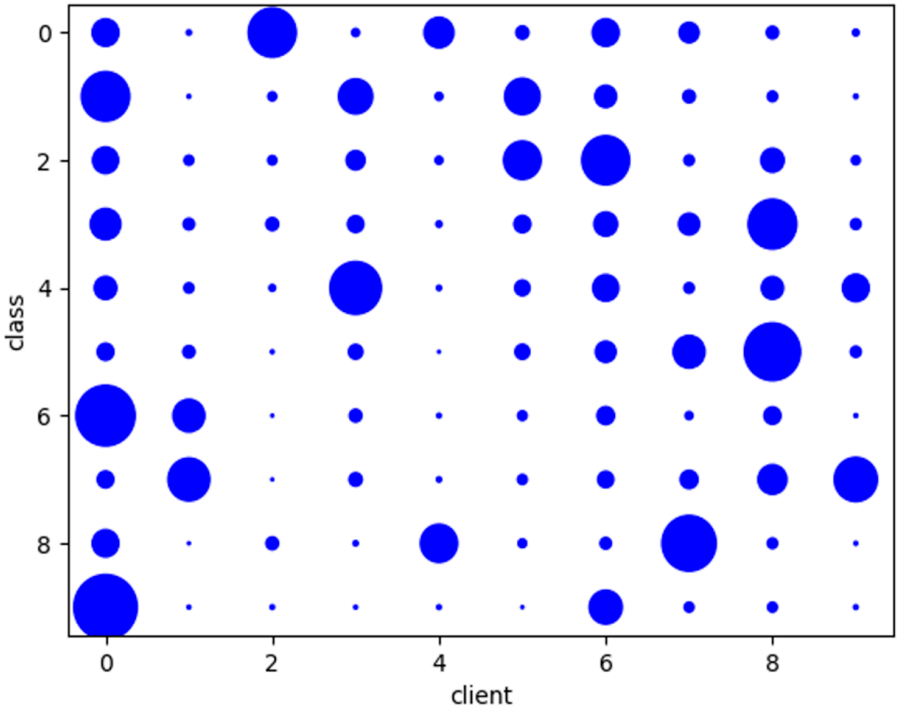


Figure 9. The statistical distribution of the top-1 feature space density's entropy.

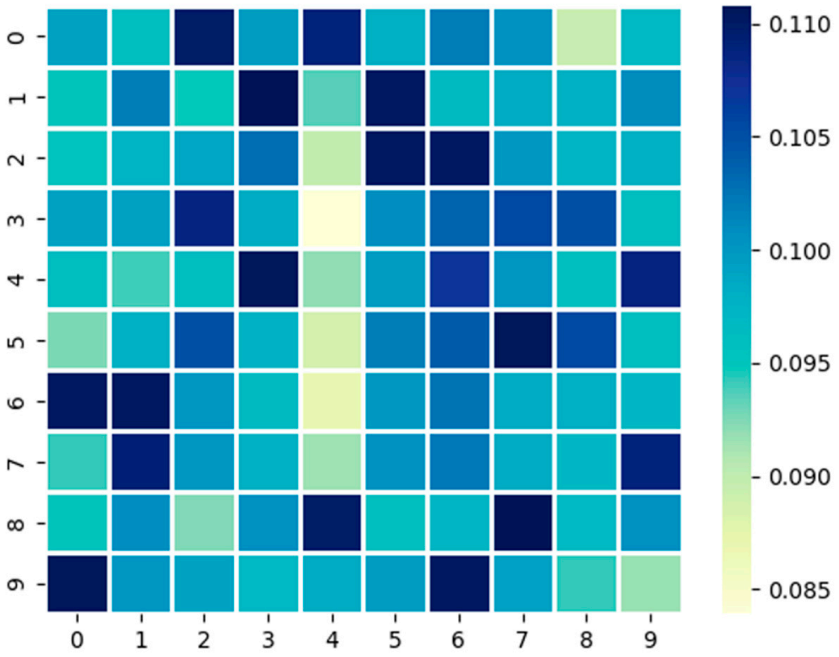


Figure 10. The mean of the entropy values obtained in Figure 8.

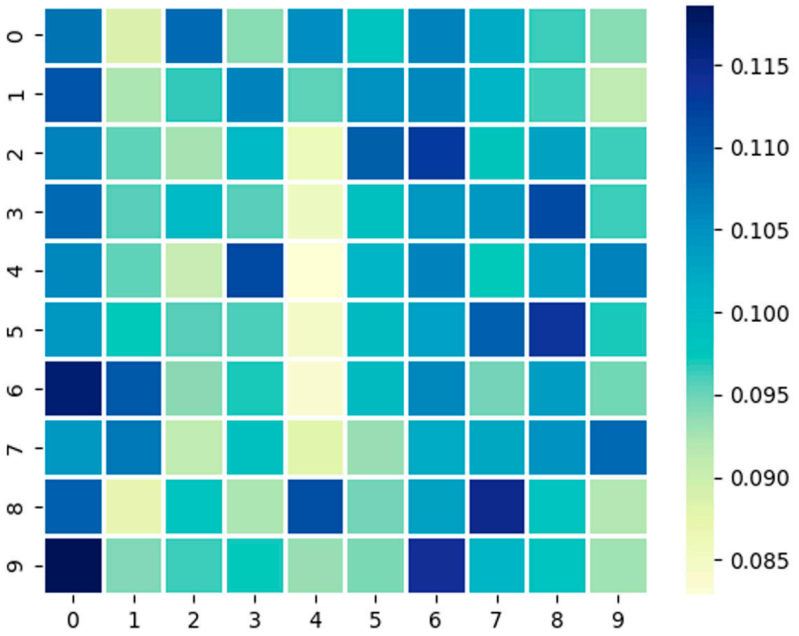


Figure 11. The mean of the entropy values obtained in Figure 9.

From the observations of the distributions and the mean values depicted in Figures 8–11, it is justified that both the proposed normalization function and the feature space density method enhance the classification performance under data and model uncertainties. Moreover, if we focus on the issue of distribution uncertainty in federated learning, the results associated with the feature space become more informative. That is, we can determine the correct classes from the darkness of the colors in Figure 11 as much more accessible than in Figure 10. This explains why our FedUA ultimately uses the feature space density method.

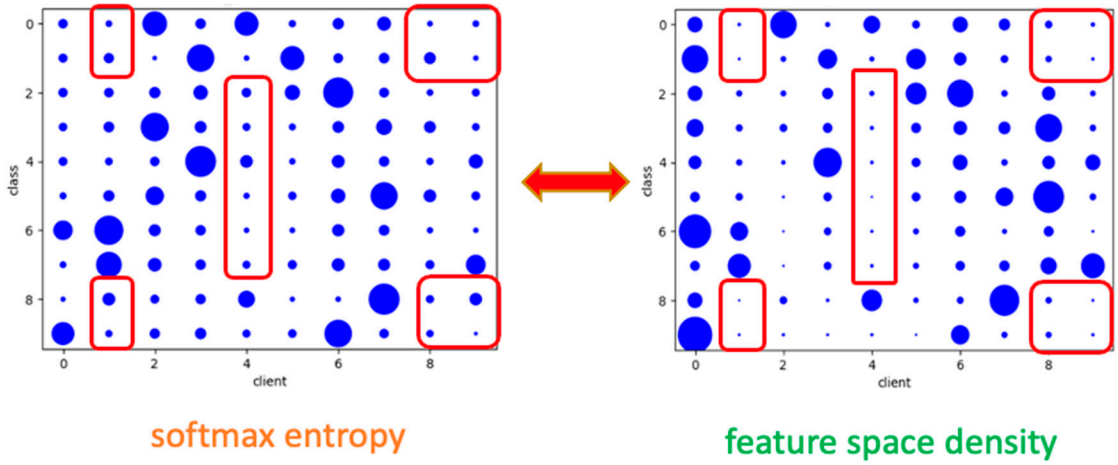


Figure 12. The sparsity comparison between the top-1 SoftMax outputs' entropy and the top-1 feature space density's entropy.

4.2.2. Performance Comparisons among the Benchmarked Works

(a) **Learning Behaviors of the Different FL-schemes on Non-IID Data**

As addressed in Section 4.1. we implement FedUA based on the pre-described settings and use the step method [20] to find its effectiveness on non-IID data. Of course, we investigate the learning behaviors of competing aggregation approaches for comparison purposes. Figure 13 shows the learning curves of FedUA, FedDF, and FedAvg, where the vertical axis denotes the accuracy percentage and the horizontal axis stands for the round number.

Dataset	**CIFAR-10**
Labeled / Unlabeled	**40,000/10,000**
Non-IID Data	**Step Method**
Rounds	**40**
#Clients	**10**
Reporting Fraction	**1.0**
Network Model	**ResNet-32**
Epoch (client/server)	**20/20**
Batch Size (client/server)	**32/128**
Loss(client/server)	**CE/KL**

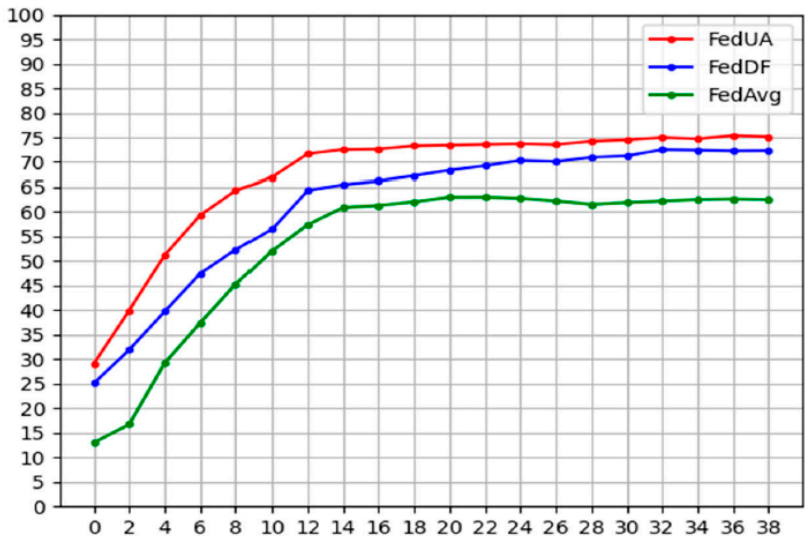


Figure 13. The learning curves of FedUA, FedDF, and FedAvg (horizontal-axis: the round number and vertical-axis: the accuracy percentage).

As shown in Figure 13, in the early stage, benefiting from knowledge distillation, the accuracy of the global models of FedUA and FedDF was significantly better than that of FedAvg. The performance-enhancing speed of the three is close, and it begins to slow down and converges to an upper limit after 14 rounds. In the end, both FedUA and FedDF outperform FedAvg, and the accuracy of FedUA is about 2–3% better than FedDF when the data distribution is non-IID.

(b) **The Impact of Different Non-IID Data Partitions**

To dive into the effects of non-IID data on various FL schemes, in this subsection, we examine the performances of FedUA, FedDF, and FedAvg under different non-IID settings. Figure 14, from left to right, shows the other non-IID data corresponding to the step method, the Dirichlet with $\alpha = 0.1$, and the Dirichlet with $\alpha = 0.5$. Table 2 compares the obtained classification accuracy among the benchmarked outcomes. The boldfaced items in Table 2 show that FedUA performs the best among the three.

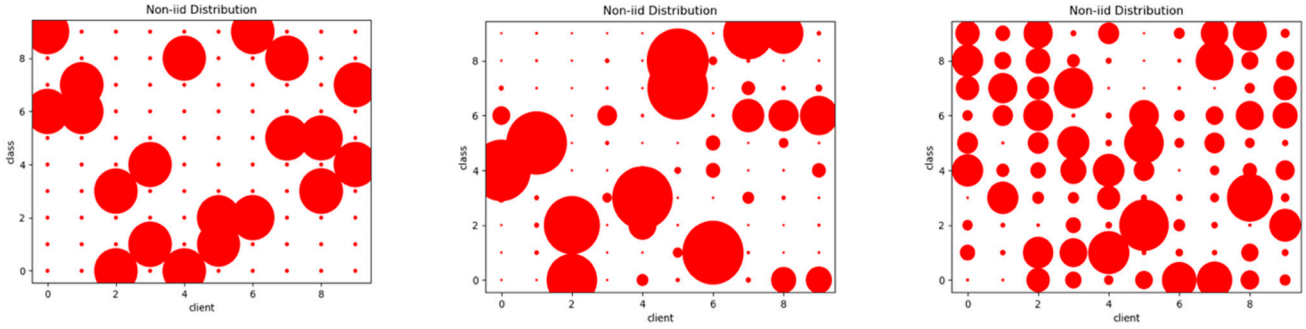


Figure 14. The distributions under different non-IID data settings.

Table 2. The performances of FedAvg, FedDF, and FedUA under different non-IID data. Settings illustrated in Figure 14. (Fed-α-Step and FedUA-FedAvg denote the Percentages of Accuracy Change vs. Parameter α when we take the Step method's and FedAvg's results as the reference, respectively.)

Classification Accuracy vs. Parameter α			
	Step Method [20]	Dirichlet ($\alpha = 0.1$)	Dirichlet ($\alpha = 0.5$)
FedAvg	62.6 ± 0.23 1 1	59.6 ± 1.03 -4.7% 1	80.1 ± 0.45 $+28\%$ 1
FedDF FedDF-α-Step FedDF-FedAvg	72.3 ± 0.49 1 15.5 %	64.4 ± 0.93 -11% 8.0%	82.8 ± 0.47 $+14.5\%$ 3.4%
FedUA FedUA-α-Step FedUA-FedAvg	**74.8 ± 0.45** **1** **19.5%**	**65.3 ± 0.78** **-12.7%** **9.6%**	**83.4 ± 0.24** **$+11.5\%$** **4.1%**

More specifically, from Figure 14, we observed a more severe data imbalance and distribution uncertainty between clients under the Dirichlet ($\alpha = 0.1$) setting. The advantages of the FedUA core method are less prominent than the step method, which is only about 1.6% growth in accuracy compared with the counterpart of FedDF (cf. Table 2). Nevertheless, there is still a meaningful improvement in knowledge distillation compared to FedAvg. For the settings under Dirichlet ($\alpha = 0.5$), the client's data is closer to IID, and both FedUA and FedDF behaved normally and better than FedAvg. The reason is that under the data segmentation of Dirichlet, a more serious data imbalance and more complex feature space density patterns are derived, resulting in more difficulty in model uncertainty and distribution uncertainty estimation. Fortunately, considering the real-world usage of federated learning nowadays, the local data distribution between devices should tend to the step method, which embraces the adaptation of the proposed FedAU in federated machine learning.

(c) **The Effects of Limited Allowable Communication Capacity**

Finally, we consider the limited communication cost scenario faced by federated learning practices. Finding enough computational resources and large datasets to conduct accurate and concrete experiments is challenging in academia. To face this reality,

we designed our simulations concerning the effects of limited allowable communication capacity by adjusting the clients' participation ratio (denoted by C) uploaded to the server for each round. We set different participation ratios and observed the corresponding results (cf. Table 3).

Table 3. The performances of FedAvg, FedDF, and FedUA under different participation ratios. (Accur-drop denotes the percentage of accuracy drop by taking C = 1.0 as the reference.)

Classification Accuracy vs. Participation Ratio			
	C = 1.0	C = 0.7	C = 0.4
FedAvg	62.6 ± 0.23	61.3 ± 0.35	58.1 ± 0.26
Accur-drop	1	2.1%	7.2%
FedDF	72.3 ± 0.49	68.1 ± 0.61	63.8 ± 1.03
Accur-drop	1	5.9%	11.8
FedUA	74.8 ± 0.45	71.8 ± 0.56	68.3 ± 0.85
Accur-drop	1	4.0%	8.7%

From Table 3, our proposed FedUA performed the best concerning absolute classification accuracy. Moreover, Table 3 also indicates that the accuracy drop of FedAvg does not decline significantly if the participation ratio is lowered below the 0.7 settings, but that of FedDF declines the most of the three (ranges from 6% to 12%, approximately). While FedUA behaves in between with an accuracy drop ranging from 4% to about 9%.

When the participation ratio of the native distillation-based federated learning decreases, the function of the teacher model is insufficient. That is, knowledge deficiency has occurred, which may make the unlabeled data used in the server aggregation stage find no correct learning objectives. As a result, FedDF relies heavily on the client to participate in the aggregation stably. However, by introducing the sample evaluation and uncertainty measurement, FedUA somehow mitigates the impact of the above shortages and avoids the damage caused to the student model when meaningless or even erroneous learning mode scenarios occur. We can justify the above arguments from the experimental results obtained in Table 3.

5. Discussions and Conclusions

5.1. Current Progress in FL Dealing with Non-IID Data

Regarding the challenges faced in FL, people in different fields will have different perspectives. This paper focuses on the countermeasures we can take when the data distributions in FL are heterogeneous. One of the anonymous reviewers suggested lots of related literature [19,30–35] and asked us to make some focused summaries and comparisons among them. Therefore, before concluding our work, this section briefly summarizes various researchers' current efforts.

Smietanka et al. [30] briefly surveyed privacy-preserving techniques and applications concerning FL. Technique-wise, three kinds of data access-related security protection methods were discussed: differential privacy, secure multiparty computation, and homomorphic encryption. At the same time, FL-related applications in Google Gboard, Health, Retail, Finance, and Insurance were addressed as illustrative examples.

To combine the advantages of cloud-based and edge-based FL for speeding up the model training and improving the communication-computation trade-offs, ref. [31] proposed a hierarchical FL architecture using multiple edge servers to perform partial model aggregation before communicating with the cloud parameter server. Empirical experiments verified the analysis and demonstrated the benefits of this hierarchical architecture in different data distribution scenarios. In other words, introducing the intermediate edge servers can simultaneously reduce the end devices' model training time and energy consumption compared to cloud-based federated learning. However, ref. [31] ignored the effects of

heterogeneous communication conditions and computing resources on different clients. Moreover, the performance-sensitive parameters need to be tuned empirically.

The authors of [32] pointed out that to train statistical models in a massive and heterogeneous network, naively minimizing an aggregate loss function may only benefit some involved devices. To face this shortage, ref. [32] proposed the so-called q-fair federated learning (q-FFL), that encourages a fairer (specifically, more uniform) accuracy distribution across devices in FL networks. Moreover, experimental results showed that with the aid of the newly devised aggregation mechanism q-FedAvg, q-FFL outperforms existing benchmarks regarding fairness, flexibility, and efficiency. Nevertheless, q-FFL increases the accuracy of poor-performing devices by sacrificing better-performing ones. This approach may not be suitable for performance-critical applications. Moreover, we need to determine the control parameter in advance again.

Tao et al. [19] proposed using ensemble distillation for model fusion, i.e., training the central classifier through unlabeled data on the outputs of the models from the clients. The authors claimed that the knowledge distillation technique would mitigate privacy risk and cost to the same extent as the baseline FL algorithms, but allowed flexible aggregation over heterogeneous client models that differed in size, numerical precision, or structure. They justified their claims through extensive empirical experiments on various CV/NLP datasets (CIFAR-10/100, ImageNet, AG News, SST2) and settings (heterogeneous models/data) by showing that the server model can be trained much faster, requiring fewer communication rounds than any existing FL technique known to them. Actually, ref. [19] inspired our work a lot.

Giovanni Paragliola and Antonio Coronato contributed a series of three papers [33–35] founded on the same kernel skills, targeting reducing communication costs in a federated healthcare environment. The inputs of the learning system were ECG waveforms of patients with various levels of risk associated with hypertension. The proposed FL framework comprised different learning strategies (varying in the numbers of cascaded dense layers and shared parameters).

To reduce the required communication costs in FL, ref. [33] presented an FL algorithm, TFedAvg, to train a time series (TS)-based model for the early identification of the level of risk associated with patients with hypertension in a federated healthcare environment. The primary framework of [33] consisted of two learning strategies, The FullNet Strategy and the PartialNet Strategy, *for which* TFedAVG exploits the whole model and a portion of the model to both guarantee the privacy and security of healthcare data, and reduce the communication costs between clients and aggregation server, respectively. Under three split local datasets conditions, ref. [33] presented two different settings concerning the types of data distribution across the regional nodes: (1) an IID setting where each node had 33% of the total samples, (2) a non-IID setting in which one of the nodes had 50% of the total samples while the other two nodes only had 25% each. Experimental results showed that the proposed approach improved from 3.01% to 11.09% in terms of classification accuracy and with a reduction of about 34% in terms of communication costs compared to the benchmarked works. Another contribution from [33] came from its summarization and comparative analyses of recent FL-related research statuses: Table 1 of [33] summarizes the studies on federated learning published between 2018 and 2022 regarding applications, adopted approaches, pros, and cons.

Yoshida et al. [6] continued and extended the discussion initiated in [33] concerning reducing the communication overhead with a further analysis, evaluating the trade-off between the performance and the communication costs. Such an analysis suggested a new learning strategy (LS) to reduce the total number of parameters shared during the FL process. The basic idea of [34] was to exploit subparts of the model in [33] by measuring the contribution of a subset of layers defining an ML model during the training process instead of the whole set of layers. To estimate the weight and the contribution of each layer, ref. [6] defined seven different learning strategies (LSs) aimed at selecting which parameters to transmit to the central server for the aggregation process, such that a trade-off between the

requirement to bring down communication costs and the need to guarantee the highest classification performance could be reached. Compared with Google's FedAvg algorithm, experimental results show that the accuracy of the approach proposed in [34] ranges from 89.25% to 96.6%. In comparison, the improvements in reducing communication overheads range from 95.64% to 6%.

The catastrophic forgetting (CF) phenomenon occurs during an ML training process when the characteristics or distribution of new input instances differ significantly from previously observed ones. CF-induced new information may overwrite the previously learned knowledge of a neural network. A similar situation might occur in FL when the local data of each client cannot be considered representative of the overall data distribution due to class imbalance, distribution imbalance, and size imbalance, that causes the well-known non-IID data challenge to FL. By successfully transferring the problem of analyzing the occurrence of CF in FL as the analysis of the DNNs' training in a federated environment when dealing with non-stationary data, ref. [35] extended the use case scenario in [6] for evaluating the nature of CF events, and provided a quantification of when and how a CF event may happen during an FL process. Finally, the experimental results in [35] depicted an improvement in accuracy ranging from 2% to 28% among local clients affected by a CF event.

5.2. Conclusions and Possible Contributions of This Work

Federated learning, constrained by security and communication costs, has flourished recently. As the above section addresses, obtaining a standard solution for various and complex Non-IID types is still challenging and worthy of further exploration.

In practice in the past, the federated learning architecture that adopted knowledge distillation usually caused incomplete interclass relationships learned by the local model due to the imbalance of local training data, which in turn made the global model learning in the aggregation stage preliminary. Therefore, we started from the uncertainty analysis of DNNs, evaluated their effects on FL, and proposed a new architecture for model aggregation. The proposed scheme improves FL's performance by combining the knowledge distillation and the DNN's uncertainty quantification methods. A series of experiments on the image classification task confirms that our proposed model aggregation scheme can effectively solve the problem of non-IID data, especially when the affordable transmission cost is limited.

The possible contributions of our work can be summarized as follows:

1. We built an effective, adaptable aggregation scheme to deal with the inhomogeneity of client side models based on the proposed quantifiable network uncertainty of the uploading client;
2. Based on the evaluated sample quality, we introduced an effective sample sieve scheme to the server to suppress the influences of data uncertainty and improve the learning efficiency;
3. As a knowledge distillation aggregation architecture, our work can effectively separate the information of uncertainty and the inter-class relationships. This separation helps solve the non-IID data issue and provides a good learning performance while limiting the transmission cost;
4. Through a series of experiments on the image classification task, we confirmed that the proposed model aggregation scheme could effectively solve the problem of non-IID data, especially when the affordable transmission cost is limited.

In summary, in handling the problem of Non-IID, we hope that the uncertainty measurement and sample evaluation we propose can help consider real-world user data. They provide more information in the aggregation stage and make learning more effective. However, our current discussion only applies to the task of image classification. Moreover, lacking enough computing resources and practical use cases are difficult to find, we have yet to be able to experiment with more complex image datasets. Nevertheless, it may be possible to combine the advantages of the uncertainty method and distillation-based

federated learning to create different collaboration models, that will be the main direction of our future efforts.

Author Contributions: Formal analysis, S.-M.L.; Funding acquisition, J.-L.W.; Investigation, S.-M.L. and J.-L.W.; Methodology, S.-M.L.; Project administration, J.-L.W.; Resources, J.-L.W.; Software, S.-M.L.; Supervision, J.-L.W.; Writing—original draft, S.-M.L.; Writing—review & editing, S.-M.L. and J.-L.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Minister of Science and Technology, Taiwan MOST 109-2218-E-002-015 and MOST 111-2221-E-002-134-MY3. And The APC was funded by MPDI.

Data Availability Statement: Not applicable.

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

References

1. McMahan, B.; Moore, E.; Ramage, D.; Hampson, S.; Arcas, B. Communication-Efficient Learning of Deep Networks from Decentralized Data. In *Artificial Intelligence and Statistics*; PMLR. 2017. Available online: https://proceedings.mlr.press/v54/mcmahan17a/mcmahan17a.pdf (accessed on 1 April 2023).
2. Kairouz, P.; McMahan, H.B.; Avent, B.; Bellet, A.; Bennis, M.; Bhagoji, A.N.; Bonawitz, K.; Charles, Z.; Cormode, G.; Cummings, R.; et al. Advances and Open Problems in Federated Learning. *Found. Trends Mach. Learn.* **2021**, *14*, 1–210. [CrossRef]
3. Zhao, Y.; Li, M.; Lai, L.; Suda, N.; Civin, D.; Chandra, V. Federated learning with non-IID data. *arXiv* **2018**, arXiv:1806.00582. [CrossRef]
4. Xiao, P.; Cheng, S.; Stankovic, V.; Vukobratovic, D. Averaging Is Probably Not the Optimum Way of Aggregating Parameters in Federated Learning. *Entropy* **2020**, *22*, 314. [CrossRef] [PubMed]
5. Zhu, H.; Xu, J.; Liu, S.; Jin, Y. Federated learning on non-IID data: A survey. *Neurocomputing* **2021**, *465*, 371–390. [CrossRef]
6. Yoshida, N.; Nishio, T.; Morikura, M.; Yamamoto, K.; Yonetani, R. Hybrid-FL for wireless networks: Cooperative learning mechanism using non-IID data. In Proceedings of the ICC 2020-2020 IEEE International Conference on Communications (ICC), Dublin, Ireland, 7–11 June 2020; pp. 7–11. [CrossRef]
7. Duan, M.; Liu, D.; Chen, X.; Tan, Y.; Ren, J.; Qiao, L.; Liang, L. Astraea: Self-Balancing Federated Learning for Improving Classification Accuracy of Mobile Deep Learning Applications. In Proceedings of the 2019 IEEE 37th International Conference on Computer Design (ICCD), Abu Dahbi, United Arab Emirates, 17–20 November 2019; pp. 246–254. [CrossRef]
8. Ghosh, A.; Hong, J.; Yin, D.; Ramchandran, K. Robust federated learning in a heterogeneous environment. *arXiv* **2019**, arXiv:1906.06629.
9. Ghosh, A.; Chung, J.; Yin, D.; Ramchandran, K. An Efficient Framework for Clustered Federated Learning. *IEEE Trans. Inf. Theory* **2022**, *68*, 8076–8091. [CrossRef]
10. Li, T.; Sahu, A.; Zaheer, M.; Sanjabi, M.; Talwalkar, A.; Smith, V. Federated optimization in heterogeneous networks. *Proc. Mach. Learn. Syst.* 2020; 2, 429–450.
11. Hsu, T.-M.H.; Qi, H.; Brown, M. Measuring the effects of non-identical data distribution for federated visual classification. *arXiv* **2019**, arXiv:1909.06335.
12. Wang, K.; Mathews, R.; Kiddon, C.; Eichner, H.; Beaufays, F.; Ramage, D. Federated evaluation of on-device personalization. *arXiv* **2019**, arXiv:1910.10252.
13. Arivazhagan, M.G.; Aggarwal, V.; Singh, A.; Choudhary, S. Federated learning with personalization layers. *arXiv* **2019**, arXiv:1912.00818.
14. Smith, V.; Chiang, C.-K.; Sanjabi, M.; Talwalkar, A. Federated multi-task learning. *Adv. Neural Inf. Process. Syst. NeurIPS* **2017**, *30*. Available online: https://papers.nips.cc/paper_files/paper/2017 (accessed on 1 April 2023).
15. Liu, B.; Wang, L.; Liu, M. Lifelong Federated Reinforcement Learning: A Learning Architecture for Navigation in Cloud Robotic Systems. *IEEE Robot. Autom. Lett.* **2019**, *4*, 4555–4562. [CrossRef]
16. Zhou, Y.; Pu, G.; Ma, X.; Li, X.; Wu, D. Distilled one-shot federated learning. *arXiv* **2020**, arXiv:2009.07999.
17. 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* **2018**, arXiv:1811.11479.
18. Li, D.; Wang, J. Fedmd: Heterogenous federated learning via model distillation. *arXiv* **2019**, arXiv:1910.03581.
19. Lin, T.; Kong, L.; Stich, S.; Jaggi, M. Ensemble Distillation for Robust Model Fusion in Federated Learning. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 2351–2363. [CrossRef]
20. Chen, H.-Y.; Chao, W.-L. Fedbe: Making bayesian model ensemble applicable to federated learning. *arXiv* **2020**, arXiv:2009.01974.
21. Hinton, G.; Vinyals, O.; Dean, J. Distilling the knowledge in a neural network. *arXiv* **2015**, arXiv:1503.02531.
22. Gawlikowski, J.; Vinyals, O.; Dean, J. A survey of uncertainty in deep neural networks. *arXiv* **2021**, arXiv:2107.03342.
23. Hendrycks, D.; Gimpel, K. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *arXiv* **2016**, arXiv:1610.02136.

24. Gal, Y.; Ghahramani, Z. Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. In Proceedings of the 33rd International Conference on International Conference on Machine Learning, New York, NY, USA, 19–24 June 2016; pp. 1050–1059.
25. Lakshminarayanan, B.; Pritzel, A.; Blundell, C. Simple and Scalable Predictive Uncertainty Estimation Using Deep Ensembles. *Adv. Neural Inf. Process. Syst.* **2017**, *30*. Available online: https://proceedings.neurips.cc/paper/2017/file/9ef2ed4b7fd2c810847ffa5fa85bce38-Paper.pdf (accessed on 1 April 2023).
26. Ovadia, Y.; Fertig, E.; Ren, J.; Nado, Z.; Sculley, D.; Nowozin, S.; Dillon, J.V.; Lakshminarayanan, B.; Snoek, J. Can You Trust Your Model's Uncertainty? Evaluating Predictive Uncertainty under Dataset Shift. *Adv. Neural Inf. Process. Syst.* **2019**, *32*. Available online: https://proceedings.neurips.cc/paper/2019/file/8558cb408c1d76621371888657d2eb1d-Paper.pdf (accessed on 1 April 2023).
27. Guo, C.; Pleiss, G.; Sun, Y.; Weinberger, K. On calibration of modern neural networks. In *International Conference on Machine Learning*; PMLR. 2017. Available online: https://proceedings.mlr.press/v70/guo17a/guo17a.pdf (accessed on 1 April 2023).
28. Mukhoti, J.; Kirsch, A.; van Amersfoort, J.; Torr, P.H.S.; Gal, Y. Deep Deterministic Uncertainty: A Simple Baseline. *arXiv* **2021**, arXiv:2102.11582.
29. Gal, Y.; Islam, R.; Ghahramani, Z. Deep bayesian active learning with image data. In *International Conference on Machine Learning*; PMLR. 2017. Available online: https://proceedings.mlr.press/v70/gal17a/gal17a.pdf (accessed on 1 April 2023).
30. Śmietanka, M.; Pithadia, H.; Treleaven, P. Federated Learning for Privacy-Preserving Data Access. 2020. Available online: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3696609 (accessed on 1 April 2023).
31. Lumin, L.; Zhang, J.; Song, S.; Letaief, K. Client-Edge-Cloud Hierarchical Federated Learning. In Proceedings of the IEEE International Conference on Communications (ICC, IEEE), Dublin, Ireland, 7–11 June 2020. [CrossRef]
32. Tian, L.; Sanjabi, M.; Beirami, A.; Smith, V. Fair Resource Allocation in Federated Learning. ICLR 2020. *arXiv* **2019**, arXiv:1905.10497.
33. Paragliola, G.; Coronato, A. Definition of a novel federated learning approach to reduce communication costs. *Expert Syst. Appl.* **2022**, *189*, 116109. [CrossRef]
34. Paragliola, G. Evaluation of the trade-off between performance and communication costs in federated learning scenario. *Futur. Gener. Comput. Syst.* **2022**, *136*, 282–293. [CrossRef]
35. Paragliola, G. A federated learning-based approach to recognize subjects at a high risk of hypertension in a non-stationary scenario. *Inf. Sci.* **2023**, *622*, 16–33. [CrossRef]