# Enabling Scalable and Trustworthy Collaborate Learning Systems

**Huancheng Chen**
Adviser: Haris Vikalo
Department of Electrical and Computer Engineering
University of Texas at Austin

## 1 Background

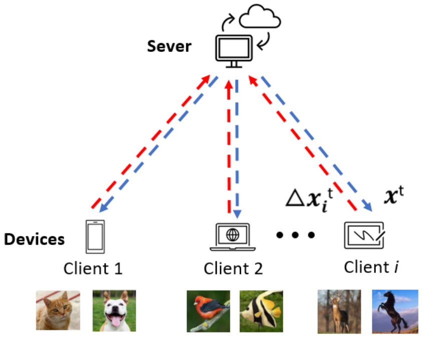


Figure 1: Example of FL workflow.

Modern distributed networks of devices such as mobile phones, wearables and smart home devices produce a vast amount of data used by machine learning models for various prediction/inference tasks. As the computational capabilities of these devices grow, along with concerns about privacy, there is a growing interest in locally storing data and moving network computation to the edge. This has motivated *federated learning (FL)* [1], a popular learning paradigm that enables privacy-preserving collaborative training of machine learning (ML) models across a number of devices by avoiding the need to collect private data stored at those devices. The canonical problem in federated learning involves the task of learning a *universal, global* statistical model under coordination of a central server. Typically, the goal is to minimize the objective function

$$\min_{\mathbf{x}} F(\mathbf{x}) \triangleq \sum_{k=1}^{N} p_k F_k(\mathbf{x}), \tag{1}$$

where $N$ is the total number of devices (referred to as *clients* in the remainder of this report), $\mathbf{x}$ denotes parameters of the global model, $F_k(\mathbf{x})$ is the loss (empirical risk) of the model on $k$-th client's data $\mathcal{B}_k$, and $p_k$ denotes the weight assigned to client $k$ ($\sum_{k=1}^{N} p_k = 1$). At each global round, only the intermediate model updates are communicated to the central server for aggregation while each clent' data remains private.

A large number of ML tasks in computer visions (CV) and natural language processing (NLP) have been adapted to the federated learning framework, demonstrating the capability of FL to produce highly accurate models by aggregating knowledge from diverse sources. However, several main challenges including **(1) statistical heterogeneity**, **(2) expensive communication**, **(3) systems heterogeneity** and **(4) safety** in distributed learning, adversely affect the performance of FL systems and make it difficult to deploy FL frameworks in realistic settings. The main focus of my research is to develop novel FL algorithms addressing the above four challenges and build scalable and trustworthy collaborate learning systems. In the following, I will shortly introduce my recent work on these topics and outline future research agenda.

## 2 Research Projects

### 2.1 Learning Accurate Global Model in Federated Learning with Statistical Heterogeneity

An early FL method, FedAvg [1], performs well in the settings where the devices train on independent and identically distributed (IID) data. However, compared to the IID scenario, training on non-IID data under statistical heterogeneity is detrimental to the convergence speed, variance and accuracy of the learned model. Figure 2 illustrates *objective drift*[2] in non-IID FL manifested through large

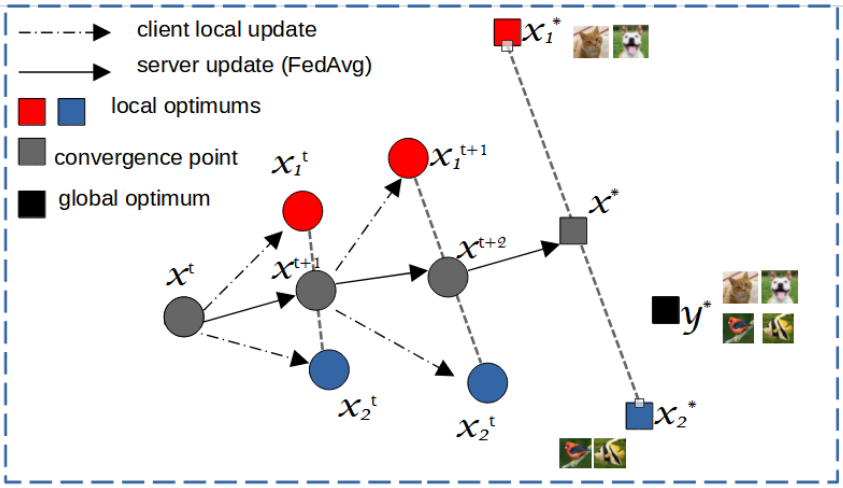


Figure 2: *objective drift*: the local optimal model of client 1 may be very distant from the local optimal model of client 2. Nevertheless, the server that deploys FedAvg still forms the global model by simply averaging the collected local models. Therefore, the global model converges toward the average of the two local optimal models $\mathbf{x}^*$ instead of the true global optimal model $\mathbf{y}^*$.

differences between local models trained on substantially different data distributions. Essentially, objective drift is caused by overfitting the local models due to class imbalance. We studied two strategies aiming to tackle statistical heterogeneity in FL: (1) data augmentation by synthesizing artificial data; (2) adding regularization terms to mitigate objective drift in local training.

In our work [3], we proposed framework FedDMPS where each client utilizes Variational Auto-Encoder (VAE) to generate synthetic data to enrich local dataset and thus ameliorate the detrimental effects of non-iid data distributions. In particular, each client trains a local VAE model (composed of encoder, classifier and decoder) with its local private (potentially class-imbalanced) data. The encoder of the local VAE model is able to extract data representations of raw data in the latent space and compute class-wise data representations by average. The server matches the pairs of clients having complementary local datasets and facilitates differentially-private[4] sharing of class-wise data representations; the clients then deploy the decoder of VAE model to reconstruct artificial data based on these shared data representations. More details of FedDMPS's workflow can be found in Figure 3.

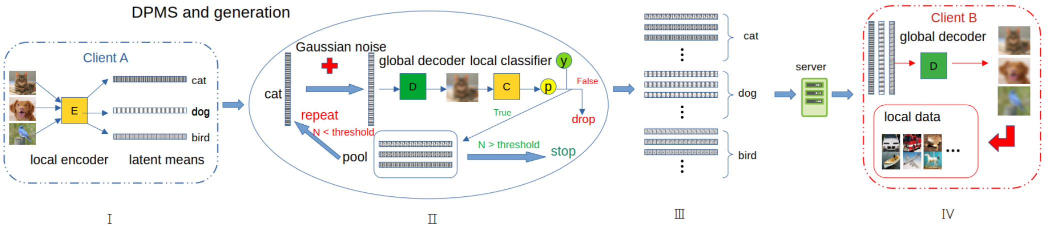


Figure 3: FedDPMS and synthetic data generation. The four parts of the figure depict: (1) finding data representation of raw data via a local encoder; (2) creating noisy means (by adding Gaussian noise to the class-wise means of data representations) and filtering out unusable ones with the help of a local classifier; (3) uploading usable noisy class-wise means to the server; (4) a benefiting client utilizing the decoder to generate synthetic data from the received noisy class-wise means, expanding its local dataset.

Although FedDPMS outperforms state-of-the-art Federated Learning methods on image classification tasks with varied levels of heterogeneity across clients, it does requires additional computation and memory resources (almost double) to train a decoder for each client. In another line of our work, FedHKD [5], we continue to exploit the shared class-wise data representations as extra information to regularize client's local objective function without introducing significant computation and memory overhead. In particular, we apply the ideas from knowledge distillation [6], improving it in FL settings by removing the need for a public dataset previously required at the server. To be specific, each client $k$ learns a feature extractor $R_k(\cdot)$ and a classifier $G_k(\cdot)$ and uses them to compute the local knowledge – average class-wise features and the corresponding predictions on clients' data – as follows:

$$\mathbf{h}_k^c = \frac{1}{N_k^c}\sum_{\mathbf{s}_i\sim\mathcal{B}_k^c} R_k(\mathbf{s}_i) + \mathcal{N}(0,\sigma_k^2), \forall c\in[C] \tag{2}$$

$$\mathbf{q}_k^c = \frac{1}{N_k^c}\sum_{\mathbf{s}_i\sim\mathcal{B}_k^c} G_k(R_k(\mathbf{s}_i)), \forall c\in[C], \tag{3}$$

where $N_k^c$ denotes the number of samples with label $c$ in client $k$'s local dataset $\mathcal{B}_k$; $\mathcal{B}_k^c$ is a subset of $\mathcal{B}_k$ where all samples have label $c$; $C$ is the number of classes in the classification task; $\sigma^2$ is the predetermined variance of differential privacy (DP) inducing noise [4] that promotes privacy. $\mathbf{h}_k^c$ and $\mathbf{q}_k^c$ are computed and transmitted to the server as the local knowledge of client $k$, upon which the server aggregates all collected local knowledge into global knowledge defined as

$$\mathcal{H}^c = \sum_{k=1}^N p_k\mathbf{h}_k^c, \quad \mathcal{Q}^c = \sum_{k=1}^N p_k\mathbf{q}_k^c \tag{4}$$

The entire procedure is shown in Fig. 4.

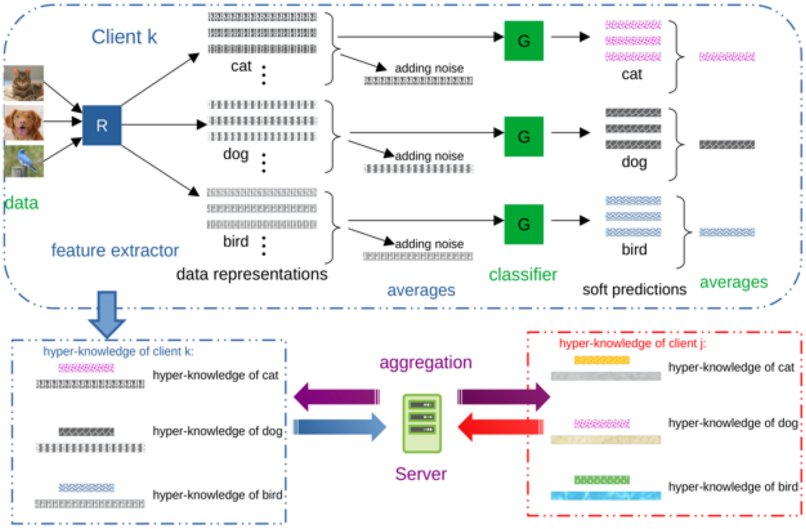


Figure 4: Procedure of computing local knowledge and aggregation.

Following the aggregation at the server, the global knowledge is sent to the clients participating in the next FL round to assist in local training. In particular, given data samples $(\mathbf{x}_i, y_i)\sim\mathcal{B}_k$, the loss function of client $k$ is formed as

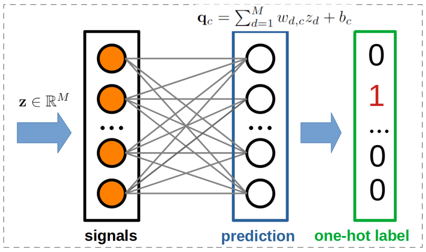
$$\mathcal{L}_k = \mathcal{L}_{\text{task}} + \lambda\frac{1}{C}\sum_{c=1}^C \|G_k(\mathcal{H}^c)-\mathcal{Q}^c\|_2 + \gamma\frac{1}{|\mathcal{B}_k|}\sum_{\mathbf{s}_i\sim\mathcal{B}_k}\|R_k(\mathbf{s}_i)-\mathcal{H}^{y_i}\|_2\,. \tag{5}$$

Note that the loss function (5) consists of three terms: the empirical risk of the original task and two regularization terms utilizing global knowledge. Essentially, the second and third terms in the loss function are proximity/distance functions. The second term is to force a local classifier to output similar soft predictions when given global data representations while the third term is to force the features extractor to output similar data representations to the average data representations in the global knowledge when given local data samples.

We analyze convergence of FedHKD and conduct extensive experiments on visual datasets in a variety of scenarios, demonstrating that FedHKD provides significant improvement in both personalized as well as global model performance compared to state-of-the-art FL methods designed for heterogeneous data settings.

### 2.2 Accelerating Non-IID Federated Learning via Client Selection

Constraints on communication resources might make it unrealistic to require regular contributions to training from all the clients in a large-scale cross-device FL system. Instead, only a fraction of clients participate in any given training round; unfortunately, this further aggravates detrimental effects of statistical heterogeneity. Selecting informative clients in non-IID FL settings is an open problem that has received considerable attention from the research community. However, efficient and effective client selection in FL remains an open challenge, motivating us develop a novel heterogeneity-guided adaptive client selection method.



$\mathbf{q}_c = \sum_{d=1}^{M} w_{d,c} z_d + b_c$
$\mathbf{z} \in \mathbb{R}^M$
0
1
...
0
0
signals
prediction
one-hot label

Figure 5: The last two network layers.

If the server were given access to clients' data label distributions, selecting clients would be relatively straightforward. However, privacy concerns typically discourage clients from sharing such information. In our latest work, HiCS-FL [7], we explored the universal property of the gradient of bias of the output layer (see Figure 5) in classification tasks and proposed a novel method to estimate the data label distribution of each client based on the local updates of the output layer's bias. In particular, we assume that the model is trained by minimizing the cross-entropy (CE) loss over one-hot labels – a widely used multi-class classification framework. Relying on the discovered gradient's property, we propose a novel method to estimate the client's data label distribution with theoretical guarantees.

Specifically, we quantify heterogeneity of clients' data by an entropy-like measure $H(\mathcal{D}^{(k)}) \triangleq -\sum_{i=1}^{C} D_i^{(k)} \ln D_i^{(k)}$, where $\mathcal{D}^{(k)}$ denotes label distribution of client $k$'s data. Under the constraint of privacy, the server does not know true $\mathcal{D}^{(k)}$; hence we approximately compute $H(\mathcal{D}^{(k)})$ using the local update $\Delta\mathbf{b}^{(k)}$ from client $k$,

$$H(\mathcal{D}^{(k)}) \approx \hat{H}(\mathcal{D}^{(k)}) = H(\text{softmax}(\Delta\mathbf{b}^{(k)}, T)), \tag{6}$$

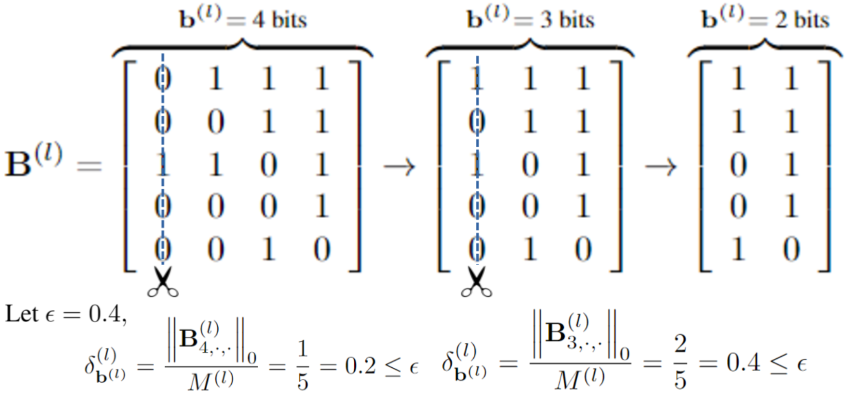
where $\text{softmax}(\Delta\mathbf{b}^{(k)}, T)_i = \exp(\Delta b_i^{(k)}/T)/\sum_{c=1}^{C} \exp(\Delta b_c^{(k)}/T)$, $1 \leq i \leq C$; here $T$ is a scaling hyper-parameter (so-called *temperature*). We provided theoretical analysis and showed that the difference between $H(\mathcal{D}^{(k)})$ and $\hat{H}(\mathcal{D}^{(k)})$ is bounded. Utilizing the approximation method defined above, the server is capable of identifying clients with class-balanced datasets and gives priority to selecting those clients in order to achieve faster convergence and higher test accuracy of the global model.

### 2.3 Mixed-Precision Quantization for Federated Learning on Resource-Constrained Heterogeneous Devices

In distributed machine learning applications, devices such as mobile phones, wearables and/or smart homes often operate with heterogeneous resources. The clients with lower computation and memory budget may not afford learning local models with the same architecture as the clients with powerful resources. Approaches to customizing model architectures to clients with varied capabilities by pruning [8] and quantization [9] have been subjects of many studies in the FL community. However, existing techniques leave much to be desired. For example, quantization techniques, either Post-Training Quantization (PTQ) or Quantization-Aware Training (QAT), require a full precision model as a teacher of quantized models which can not be applied in FL directly – the clients with low resources cannot run full precision model due to the memory peak constraint. Our latest work aims to address the challenges of quantization in federated learning under heterogeneous source constraints. Specifically, we aim to develop a mixed-precision quantization (MPQ) method that takes into account limitations on each client's resources, without running a full precision model on devices.

To this end we introduce FedMPQ [10], a novel **Fed**erated learning algorithm with **M**ixed-**P**recision **Q**uantization, which enables training of quantized local models within the allocated bit-width budget.

FedMPQ first initializes local models as fixed-precision quantized networks that satisfy clients' average bit-width budget, and then converts these quantized networks into a representation that allows bit-level sparsity-promoting training. In particular, while learning local models whose parameters admit binary representation, the clients deploy a group Lasso regularization term which imposes a trade-off between the task loss and bit-sparsity. The precision of layers that end up having parameters which exhibit higher degree of sparsity is reduced to allow increasing precision of other layers, as illustrated in Figure.6. During the aggregation step, FedMPQ employs the *pruning-growing* strategy where the server aggregates clients' models (locally trained at potentially different bit-widths) to produce a full precision global model. Before transmitting the global model to a client, the bit-width of the model is adjusted to match the client's bit-width budget.



$\mathbf{b}^{(l)} = 4$ bits
$\mathbf{b}^{(l)} = 3$ bits
$\mathbf{b}^{(l)} = 2$ bits
$\mathbf{B}^{(l)} = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & 1 \\ 1 & 1 \\ 0 & 1 \\ 0 & 1 \\ 1 & 0 \end{bmatrix}$
Let $\epsilon = 0.4$,
$\delta^{(l)}_{\mathbf{b}^{(l)}} = \frac{\left\| \mathbf{B}^{(l)}_{4,\cdot,\cdot} \right\|_0}{M^{(l)}} = \frac{1}{5} = 0.2 \leq \epsilon$
$\delta^{(l)}_{\mathbf{b}^{(l)}} = \frac{\left\| \mathbf{B}^{(l)}_{3,\cdot,\cdot} \right\|_0}{M^{(l)}} = \frac{2}{5} = 0.4 \leq \epsilon$

Figure 6: Procedure of bit-level pruning.

### 2.4 Private Data Extraction in Federated Learning (in progress)

While clients' privacy is a main motivation for federated learning, recent work [11] shows the possibility of extracting private data from the model updates via gradient inversion attack. This kind of attack can be described as an inverse problem

$$\mathbf{s} = \arg\min_{\mathbf{s},y} \left\| \nabla_{\mathbf{x}} F_k(\mathbf{s}, y) - \nabla_{\mathbf{x}} F_k(\mathbf{s}^*, y) \right\|^2 , \tag{7}$$

where $\mathbf{s}^*$ is the true private data and $\nabla_{\mathbf{x}} F_k(\mathbf{s}^*, y)$ is the measurement. Although the original attack in [11] performs well only when the federated learning system involves small batch size (for instance, 2 or 4), the subsequent studies [12] improve the gradient inversion attack for larger batch sizes (e.g., 128) by recovering labels under some strong assumptions given an untrained model. To explore potential vulnerabilities of privacy in federated learning, our latest work is developing a novel gradient inversion attack performing well even given a well-trained model and under rather mild general assumptions. After evaluating the vulnerability of FL, our following step will be to propose defending mechanism to prevent private data extraction and explore the trade-off between safety and performance.

## 3 Future Directions

In addition to the topics outlined in the previous section, there are other interesting direction that I would like to explore in my research. These include:

- **Personalized Federated Learning.** In some scenarios, it is unrealistic and perhaps even meaningless to train a highly accurate global model. Instead, each client only needs to learn a accurate local model performing well on the local test data. Personalized Federated Learning (pFL) allows each client to have customized model architecture, model bitwidth and objective function for achieving a better test accuracy on specific data distribution.
- **Robust Aggregation in Federated Learning.** The vanilla Federated Learning, FedAvg, aggregates local models into the global model by naively computing their weighted average.

Such simple aggregation is vulnerable to Byzantine Clients who are providing detrimental local updates or conducting backdoor attack. It is critical to develop more robust aggregation methods to deploying FL systems in real work.

## References

[1] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. PMLR, 2017.

[2] Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. Scaffold: Stochastic controlled averaging for federated learning. In *International conference on machine learning*, pages 5132–5143. PMLR, 2020.

[3] **Huancheng Chen** and Haris Vikalo. Federated learning in non-iid settings aided by differentially private synthetic data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 5026–5035, June 2023.

[4] Cynthia Dwork. Differential privacy. In *International colloquium on automata, languages, and programming*, pages 1–12. Springer, 2006.

[5] **Huancheng Chen**, Chaining Wang, and Haris Vikalo. The best of both worlds: Accurate global and personalized models through federated learning with data-free hyper-knowledge distillation. In *The Eleventh International Conference on Learning Representations*, 2023.

[6] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[7] **Huancheng Chen** and Haris Vikalo. Accelerating non-iid federated learning via heterogeneity-guided client sampling. *arXiv preprint arXiv:2310.00198*, 2023.

[8] Sameer Bibikar, Haris Vikalo, Zhangyang Wang, and Xiaohan Chen. Federated dynamic sparse training: Computing less, communicating less, yet learning better. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 6080–6088, 2022.

[9] Amirhossein Reisizadeh, Aryan Mokhtari, Hamed Hassani, Ali Jadbabaie, and Ramtin Pedarsani. Fedpaq: A communication-efficient federated learning method with periodic averaging and quantization. In *International Conference on Artificial Intelligence and Statistics*, pages 2021–2031. PMLR, 2020.

[10] **Huancheng Chen** and Haris Vikalo. Mixed-precision quantization for federated learning on resource-constrained heterogeneous devices. *arXiv preprint arXiv:2311.18129*, 2023.

[11] Jonas Geiping, Hartmut Bauermeister, Hannah Dröge, and Michael Moeller. Inverting gradients-how easy is it to break privacy in federated learning? *Advances in Neural Information Processing Systems*, 33:16937–16947, 2020.

[12] Hongxu Yin, Arun Mallya, Arash Vahdat, Jose M Alvarez, Jan Kautz, and Pavlo Molchanov. See through gradients: Image batch recovery via gradinversion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16337–16346, 2021.