Specialized federated learning using a mixture of experts

Edvin Listo Zec [1] **Olof Mogren** [1] **John Martinsson** [1] **Leon René Sütfeld** [1] **Daniel Gillblad** [2]

Abstract

In federated learning, clients share a global model that has been trained on decentralized local client data. Although federated learning shows significant promise as a key approach when data cannot be shared or centralized, current methods show limited privacy properties and have shortcomings when applied to common real-world scenarios, especially when client data is heterogeneous. In this paper, we propose an alternative method to learn a personalized model for each client in a federated setting, with greater generalization abilities than previous methods. To achieve this personalization we propose a federated learning framework using a mixture of experts to combine the specialist nature of a locally trained model with the generalist knowledge of a global model. We evaluate our method on a variety of datasets with different levels of data heterogeneity, and our results show that the mixture of experts model is better suited as a personalized model for devices in these settings, outperforming both fine-tuned global models and local specialists.

1. Introduction

In many real-world scenarios, data is distributed over a large number of devices or across many organizations, due to privacy concerns or communication limitations. Federated learning is a framework that can leverage this data in a distributed learning setup. This allows for the use of all participating clients' compute power, with the added benefit of a large decentralized training data set, while enhancing privacy and data security.

[1]RISE Research Institutes of Sweden [2]AI Sweden. Correspondence to: Edvin Listo Zec <edvin.listo.zec@ri.se>.

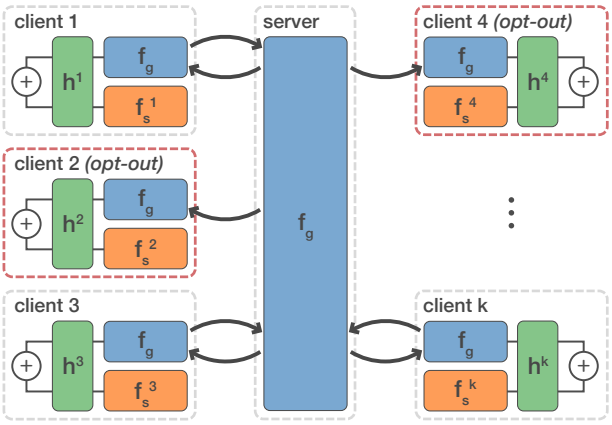


Figure 1. Federated mixtures of experts, consisting of a global model f_g and local specialist models f_s^k using local gating functions h^k. Some clients opt-out from federation, not contributing to the global model and keeping their data completely private.

For instance, in keyboard prediction for smartphones, thousands or even millions of users produce keyboard input that can be leveraged as training data. The training can ensue directly on the devices, doing away with the need for costly data transfer, storage, and immense compute on a central server (Hard et al., 2018). The medical field is another example area where data is often extremely sensitive and cannot be shared externally, thus requiring distributed and privacy-protecting approaches.

The optimization problem that we solve in a federated learning setting is

$$\min_{w \in \mathbb{R}^d} \frac{1}{K} \sum_{k=1}^{K} \mathbb{E}_{(x,y)\sim p_k} \left[\ell_k(w;\ x, y) \right] \tag{1}$$

where ℓ_k is the loss for client k and (x, y) samples from the kth client's data distribution p_k. A central server is coordinating training between the K local clients. The most prevalent algorithm for solving this optimization is the federated averaging (FEDAVG) algorithm (McMahan et al., 2017). In this solution, each client has its own client model, parameterized by w^k which is trained on a local dataset for E local epochs. When all clients have completed the training, their weights are sent to the central server where they are aggregated into a global model, parameterized by

w_g. In FEDAVG, the k client models are combined by parameter averaging, weighted by the size of their respective local datasets:

$$w_g^{t+1} \leftarrow \sum_k \frac{n_k}{n} w_{t+1}^k, \tag{2}$$

where n_k is the size of the dataset of client k and $n = \sum_k n_k$. Finally, the new global model is sent out to each client, where it constitutes the starting point for the next round of (local) training. This process is repeated for a defined number of global communication rounds.

The averaging of local models in parameter space generally works but requires some care to be taken in order to ensure convergence. (McMahan et al., 2017) showed that all local models need to be initialized with the same random seed for FEDAVG to work. Extended phases of local training between communication rounds can similarly break training, indicating that the individual client models will over time diverge towards different local minima in the loss landscape. Similarly, different distributions between client datasets will also lead to divergence of client models (McMahan et al., 2017).

Depending on the use case, however, the existence of local datasets and the option to train models locally can be advantageous: specialized local models, optimized for the data distribution at hand may yield higher performance in the local context than a single global model, although typically at the cost of generalization performance. Keyboard prediction, for example, based on a global model may represent a good approximation of the population average, but could provide a better experience at the hands of a user when biased towards their individual writing style and word choices. This raises an important question – when is a global FL-trained model better than a specialized local model? A specialist would be expected to perform better than a global generalist in a highly non-iid setting, whereas the global generalist would be expected to perform better in an iid setting.

There are several ways client distributions can be non-identical. The conditional distributions $P_i(x|y)$ on all clients i may be the same, but the marginal distributions $P_i(x)$ may vary (covariate shift) or $P_i(y)$ may vary (prior probability shift). Further, if the marginal distribution $P(y)$ is the same on all clients, the conditional $P_i(x|y)$ may vary (same label, different features) or $P(x)$ is the same, but the conditional $P_i(y|x)$ varies (same features, different label). In this work we study non-identical distributions in the form of prior probability shift, although we hypothesize that our proposed method can handle other distributional shifts as well and would be an interesting direction for future work.

To address the issue of specialized local models within the federated learning setting, we propose a general framework based on a mixture of experts (Jacobs et al., 1991). In this work we have one mixture of experts per client, each combining one local specialist model and one global model. Each client has a local *gating function* that performs a weighting of the experts dependent on the input data. First, the global model is trained using FEDAVG. This is followed by training of all clients' local specialist models, initialized with the trained federated global model. This is followed by training of the entire mixture, i.e., the local and global models as well as the gating function. A common problem with fine-tuned specialist models is that although they achieve better accuracy on local test data, they do not generalize as well as a global model. However, in our work we show that we can reach the same local accuracy on client data as a fine-tuned model, while retaining superior generalization performance.

While standard federated learning already shows some privacy enhancing properties, it has been shown that in some settings, properties of the client and of the training data may be reconstructed from the gradients communicated to the server (Wang et al., 2019). Therefore, we will work with a stronger notion of privacy in this paper. While existing solutions may be private enough for some settings, we will assume that clients requiring privacy for some of their data need this data not to have any influence on the training of the global model at all. Instead, our framework allows for a complete opt-out from the federation with some or all of the data for any client. Clients with such preferences will still benefit from the global model and retain a high level of performance on their own, skewed data distribution. This is important when local datasets are particularly sensitive, as may be the case in medical applications. Our experimental evaluations demonstrate the robustness of our learning framework with different levels of label heterogeneity in the data, and under varying fractions of opt-out clients.

2. Related work

Distributed machine learning has been studied as a strategy to allow for training data to remain on the clients, giving it some aspects of privacy, while leveraging the power of learning from bigger data and compute (Konečnỳ et al., 2016; Shokri & Shmatikov, 2015; McMahan et al., 2017; Vanhaesebrouck et al., 2017; Bellet et al., 2018). The federated averaging algorithm (McMahan et al., 2017) has been influential and demonstrated that averaging of the weights in neural network models trained separately at the clients is successful in many settings, producing a federated model that demonstrates the ability to generalize from limited subsets of data at the clients. However, it has been shown that federated averaging struggles when data is not independent and identically distributed among the clients, e.g., in the problem of prior probability shift. This illustrates the need

for client personalization within federated learning (Kairouz et al., 2019; Hsieh et al., 2020).

In general, addressing class imbalance is still a relatively understudied problem in deep learning (Johnson & Khoshgoftaar, 2019). A common approach for personalization on skewed label distributions is to first train a generalist model and then fine-tune it using more specific data. This approach is used in meta-learning (Finn et al., 2017), domain adaptation (Mansour et al., 2009), and transfer learning (Oquab et al., 2014). For the distributed setting, fine-tuning was first proposed by (Wang et al., 2019) who used federated averaging to obtain a generalist model which was later fine-tuned locally on each client, using each client's specific training data. Some work has been inspired by the meta-learning paradigm to learn models that are specialized at the clients (Jiang et al., 2019; Fallah et al., 2020). (Arivazhagan et al., 2019) combined this strategy and ideas from transfer learning with deep neural networks and presented a solution where shallow layers are frozen, and the deeper layers are retrained at every client.

(Zhao et al., 2018) propose a strategy to improve training on non-iid client data by creating a subset of data which is globally shared between all clients. (Hsu et al., 2019) show that performance degrades when client distributions shift, and propose to solve the problem via server momentum. Recent strategies have also explored knowledge distillation techniques for federated learning (Jeong et al., 2018; He et al., 2020; Lin et al., 2020), which show promising results in non-iid settings.

Mixing models. (Deng et al., 2020) proposed to combine a global model w trained using federated averaging, with a local model v with a weight α_i. To find optimal α_i they optimize $\alpha_i^* = \arg\min_{\alpha_i \in [0,1]} f_i\left(\alpha_i \boldsymbol{v} + (1-\alpha_i)\,\boldsymbol{w}\right)$ in every communication round. While this weighting scheme will balance the two models, it is unable to adapt to the strengths of the different members of the mix.

(Hanzely & Richtárik, 2020) proposed a solution that provides an explicit trade-off between global and local models by the introduction of an alternative learning scheme that does not take the full federation step at every round, but instead takes a step in the direction towards the federated average.

Mixture of experts have previously been used for learning private user models in a federated setting (Peterson et al., 2019). Although experiments are limited, the authors show that a mixture of a local and global model is more robust to differential privacy noise than a global model trained with federated averaging.

Contributions. In this work, we integrate mixture of experts into a non-iid federated setting in order to learn a personalized federated model for each client. More specifically, we leverage the strengths of a global model trained with federated averaging and a local model trained locally on each client. We show empirically on multiple datasets that our method outperforms both a locally trained model and a global federated model fine-tuned on test data from each client. Our results also show that our proposed method generalizes better than both baseline personalization methods. Further, we show our proposed method to be robust against low client participation, making it possible for clients to opt out from the global federation, and to keep the data completely private for these clients.

3. Federated learning using a mixture of experts

In this work, we present a framework for model personalization in a non-iid federated learning setting that builds on federated averaging and mixtures of experts. Our framework includes a personalized model for each client, which consists of a mixture of a globally trained model and a locally trained specialist. The local models never leave the clients, which gives strong privacy properties, while the global model is trained using federated averaging and leverages larger compute and data. In our framework, as illustrated in Figure 1, clients can choose to opt-out from the federation. This ensures complete privacy for their data, as no information from their data ever leaves the client.

Let f_g be the global model with parameters w_g. We denote the index of clients by k and the local specialist models by f_s^k with parameters w_s^k. The gating function is denoted by h^k, parameterized with w_h^k. Training in the proposed framework is divided into three main parts. First, a global model f_g is trained using federated averaging using opt-in data (see Section 3.1). Second, a local specialist model f_s^k is created for each client, initialized with the weights of the global model and fine-tuned using the local opt-in data on the client. Third, we train the mixture of experts

$$h^k(x) f_s^k(x) + (1 - h^k(x)) f_g(x). \tag{3}$$

We freeze the weights of f_g, only updating the local specialist f_s^k and the gating model h^k on each client. By freezing the weights of f_g in the last step, we ensure that the generalist knowledge in the model is not unlearned in the fine-tuning procedure.

As the mixture is trained, the two expert models f_s and f_g compete against each other, while the gating function sends an error signal guiding the winner of the two experts for every input. Over the course of this procedure, the gating function will learn to separate the input space given how well each expert perform on the task.

3.1. Opting out from federation

Users requiring high levels of privacy may not want to participate in the federation and disclose their locally trained model to a central server. For this reason, our proposed framework allows for clients to opt-out from federation. Each client may arbitrarily partition its data into a part that is not used in the federation, and a part that is. No information from the opt-out data will ever leave the client. The system will still leverage learning from this data by using it to train the local specialist model f_s^k and the gating model h^k. This is a very flexible and useful property as it allows for the use of sensitive data in training of the private local models, while transformations of it, created by some privatization mechanism (e.g. differential privacy), can be used to train the federated model.

Formally, each client dataset $\mathcal{D}^k$ is split into two non-overlapping datasets, $\mathcal{D}^k_{\mathcal{O}}$ and $\mathcal{D}^k_{\mathcal{I}}$, one of which has to be non-empty. The local model f_l^k and the gating model h^k is trained using the whole dataset $\mathcal{D}^k = \mathcal{D}^k_{\mathcal{O}} \cup \mathcal{D}^k_{\mathcal{I}}$, while the global model f_g is trained with FEDAVG using only the non-sensitive *opt-in* dataset $\mathcal{D}^k_{\mathcal{I}}$. In Figure 1 this is visualized by each client either opting-in or out all of its data. In our experiments, we assume that a client that opts out does so with its whole dataset, meaning that it puts all of its data in $\mathcal{D}^k_{\mathcal{O}}$.

3.2. Optimization problem

Step 1: Train a global model. We train the global model using FEDAVG. In other words, globally we optimize

$$\min_{w_g \in \mathbb{R}^d} \frac{1}{\left|\mathcal{D}^k_{\mathcal{I}}\right|} \sum_{k \in \mathcal{D}^k_{\mathcal{I}}} \mathbb{E}_{(x,y)\sim\mathcal{D}^k_{\mathcal{I}}} \left[\ell_k(w_g;\ x, y, \hat{y}_g)\right] \tag{4}$$

for the opt-in dataset $\mathcal{D}^k_{\mathcal{I}}$. Here ℓ_k is the loss for the global model w_g on client k for the prediction $f_g(x) = \hat{y}_g$, and $\mathcal{D}^k_{\mathcal{I}}$ is the kth clients *opt-in* data distribution.

Step 2: Train local specialists. The output model from FEDAVG is fine-tuned on each clients datasets, minimizing the local loss. We initialize the specialist model with the global parameters w_g and optimize:

$$\min_{w_s^k \in \mathbb{R}^d} \mathbb{E}_{(x,y)\sim\mathcal{D}^k} \left[\ell_k(w_s^k;\ x, y, \hat{y}_l)\right] \quad \forall k = 1, \ldots, n. \tag{5}$$

Here, ℓ_k is the loss for the prediction $\hat{y}_l = f_s^k(w_s^k;\ x)$ from the fine-tuned model on the input x and $\mathcal{D}^k$ is the kth clients dataset.

Step 3: Train local mixtures. The local mixture of experts are trained using the gating models h^k, with the prediction error given by weighing the trained models f_g and f_s^k:

$$\hat{y}_h = h^k(x) f_s^k(x) + \left(1 - h^k(x)\right) f_g(x) \quad \forall k = 1, \ldots, n. \tag{6}$$

In other words, at the end of a communication round, given f_s^k and f_g, we optimize the mixture (6):

$$\min_{w_g, w_s^k, w_h^k} \mathbb{E}_{(x,y)\sim\mathcal{D}^k} \left[\ell_k(w_g, w_s^k, w_h^k;\ x, y, \hat{y}_h)\right], \tag{7}$$

locally for every client $k = 1, \ldots, n$. Here, ℓ_k is the loss from predicting $\hat{y}$ for the label y given the input x with the model from (6) over the data distribution $\mathcal{D}^k$ of client k. We freeze the weights of w_g, and only update w_s^k and w_h^k. A summary of the method is described in Algorithm 1.

Algorithm 1

input: Models participating in FEDAVG $w_1, \ldots, w_k$, local gate w_h^k, learning rate η, decay rates β_1, β_2
Initialize $w_1, \ldots, w_k$ with the same random seed.
Initialize w_h^k.
$w_g \leftarrow$ FEDAVG$(w_1, \ldots, w_k)$ *// Train for E local epochs and G communication rounds*
for client k **do**
  Initialize specialist model $w_s^k \leftarrow w_g$.
  $w_s^k \leftarrow \text{Adam}(w_s^k, lr, \beta_1, \beta_2)$ *// Fine-tune w_g on each client k*
  Freeze global parameters w_g.
  $w_s^k, w_h^k \leftarrow \text{Adam}(w_g, w_s^k, w_h^k, lr, \beta_1, \beta_2)$ *// Train mixture of experts on client k*
end for
output: Trained mixture of experts: global model w_g, local experts w_s^k and local gating functions w_h^k.

4. Experimental setup

We use two different ways of sampling client data to simulate heterogeneous distributions. The first setup is a more generalized version of the pathological non-iid setup as described in (McMahan et al., 2017) where each client is only assigned 2 classes. The second sampling strategy is performed using the Dirichlet distribution as described in (Yurochkin et al., 2019; Hsu et al., 2019).

Datasets and models. Our experiments are carried out using two model architectures on three datasets. The dataset used are CIFAR-10 (Krizhevsky et al., 2009), Fashion-MNIST (Xiao et al., 2017), and AG News (Gulli, 2004).

The CIFAR-10 dataset consists of 60 000 32x32 color images in 10 classes, with 6000 images per class. The dataset is split into 50 000 training images and 10 000 test images.

The Fashion-MNIST dataset contains 70 000 28x28 grayscale images of Zalando clothing in 10 classes. It is split into 60 000 training images and 10 000 test images.

The AG News topic classification dataset consists of 4 classes, each of which contains 30 000 training samples and 1 900 testing samples. In total there are 120 000 training samples and 7 600 testing samples.

For CIFAR-10 and Fashion-MNIST, the specialist model f_s and the global model f_g are CNNs with the same architecture. The CNN has two convolutional layers, each with a kernel size of 5 (the first with 6 channels, the second with 16), and two fully-connected layers with 120 and 84 units, respectively, with ReLU activations. This is followed by an output layer with a softmax activation. The gating function h_k has the same architecture as f_g and f_s, but with a sigmoid activation in the output layer instead of a softmax.

For AG News, both the local and the global models consist of an embedding layer with a dimension size of 100, a bi-directional LSTM layer with 64 nodes followed by an output layer with a softmax activation. The gating function has the same architecture in this case as well, but with a sigmoid activation in the output layer. We use the Adam optimizer (Kingma & Ba, 2014) to train all models.

Pathological non-iid sampling. The first way we create a skewed non-iid dataset for each client is by constructing a subset for each client with oversampling of specific classes. Sampling is performed such that the dataset of each client contains two majority classes which together form a fraction p of the client data and the remaining classes form a fraction $(1-p)$ of the client data. We perform experiments with $p = \{0.2, 0.3, \ldots, 1.0\}$ to see what effect the degree of heterogeneity has on performance. In the extreme case $p = 1.0$, each client dataset only contains two classes in total, which is the same pathological non-iid setup as used for the MNIST dataset in (McMahan et al., 2017). A majority class fraction of $p = 0.2$ represents an iid setting for CIFAR-10 and Fashion-MNIST. For the AG News dataset which only has four classes, a fraction of $p = 0.5$ represents an iid setting.

Dirichlet distribution non-iid. The second sampling strategy is to use the Dirichlet distribution as described in (Yurochkin et al., 2019; Hsu et al., 2019). For each class we sample $\mathbf{n}_k \sim \text{Dir}_j(\alpha)$ and assign each client j a proportion of $\mathbf{n}_{k,j}$ for class k. When $\alpha \to \infty$ we have an iid setting of equal number of instances per class for each client. When $\alpha \to 0$, we have a completely non-iid setting where each client dataset only has one class in total. We form experiments with $\alpha = \{0.05, 0.1, 0.5, 1.0, 10, 100\}$.

Opt-out factor and privacy. Some users might want to opt out from participating to a global model, due to privacy reasons. These users will still receive the global model. To simulate this scenario in the experimental evaluation, we introduce an *opt-out factor* denoted by q. This is a fraction deciding the number of clients participating in the FEDAVG optimization. The clients that participate in the federated learning optimization have all their data in $\mathcal{D}_{\mathcal{I}}^k$, while the clients that opt out have all their data in $\mathcal{D}_{\mathcal{O}}^k$. $q = 0$ means all clients are participating. We perform experiments varying q, to see how robust our algorithm is to different levels of client participation. In Figure 1 we visualize how the opt-out factor can be used.

FEDAVG parameters. For CIFAR-10 and Fashion-MNIST, the training is performed using 100 clients with 100 training samples per client. A client sampling fraction of 0.05 is used, meaning that 5 clients participate in each communication round. If the opt-out fraction q is larger than 0, we change the sampling fraction such that there always are 5 clients that participate in every communication round.

For AG News, we set the number of clients to 1000, with 100 training samples per client. We use a client sampling fraction of 0.05, meaning that 50 clients participate in each communication round. If the opt-out fraction q is larger than 0, we change the sampling fraction such that there always are 50 clients that participate in every communication round.

For all datasets we set the number of communication rounds to 1250, number of local epochs to 3 and local batch size to 10. We use early stopping and validate the performance of FEDAVG on each participating client's local validation set every 50th communication round. The global model with the best mean validation loss over participating clients is returned.

Baselines. We use three different models as baselines. First, a locally trained model for every client, only trained on each clients own dataset. Second, FEDAVG. Third, the final model output from FEDAVG fine-tuned for each client on its own local data, denoted by f_s^k. We train the local model, the fine-tuned model and the mixture using early stopping for 500 epochs, monitoring local validation loss on each client and return the best performing model in each case.

Evaluation. We evaluate using both a *local* (skewed) and a *global* (balanced) test set. Each client has a local test set ($n = 500$ samples) that mirrors its local data distribution. This test set is used to measure how well a model specializes to a client. The global test set ($n = 1000$ samples) is a balanced test set (it contains the same number of data points for all classes) and is the same for all clients. We use this to measure how well a model generalizes. During evaluation, we sample 20 random clients and calculate the local and global test accuracies for all baselines and report a mean over the clients. All experiments were performed on a Tesla V100-SXM2-32GB, and all reported results are means over four runs.

5. Results and discussion

For the sake of reproducibility, all code is made available. [1]

Figure 2 shows a learning rate sweep for FEDAVG on all

[1] `Link to github repo will be made available here.`

Table 1. Accuracy on a global and local test set for all baselines on all datasets with varying majority class fractions p. Best performing specialist in bold. Opt-out fraction $q = 0$. All results reported are over four runs.

(a) CIFAR-10: Global test set

p	FedAvg	Local	Fine-tuned	Mixture
0.3	42.65	19.17	39.78	**41.57**
0.6	30.47	14.74	25.87	**26.37**
0.7	22.90	14.62	20.66	**21.45**
0.8	20.00	13.98	18.85	**19.55**
1.0	15.08	14.18	14.55	**14.66**

(b) CIFAR-10: Local test set

p	FedAvg	Local	Fine-tuned	Mixture
0.3	43.32	23.39	42.98	**43.86**
0.6	30.13	42.19	**50.97**	50.57
0.7	21.81	50.23	**54.98**	54.73
0.8	16.78	59.89	**64.54**	64.47
1.0	13.62	77.00	78.32	**78.56**

(c) Fashion-MNIST: Global test set

p	FedAvg	Local	Fine-tuned	Mixture
0.3	71.12	40.65	69.32	**70.50**
0.6	66.85	18.07	61.43	**64.82**
0.7	64.45	17.57	58.95	**62.15**
0.8	67.45	17.69	58.66	**61.53**
1.0	49.43	18.11	**23.69**	22.64

(d) Fashion-MNIST: Local test set

p	FedAvg	Local	Fine-tuned	Mixture
0.3	70.66	45.77	69.76	**70.42**
0.6	65.14	55.16	71.01	**71.18**
0.7	64.67	65.12	**74.86**	74.63
0.8	66.45	74.84	76.02	**76.70**
1.0	48.01	**94.22**	91.28	92.10

(e) AG News: Global test set

p	FedAvg	Local	Fine-tuned	Mixture
0.5	80.31	28.82	75.51	**78.03**
0.7	10.42	3.88	9.79	**10.15**
0.8	10.39	3.82	9.59	**10.01**
0.9	9.73	3.91	8.90	**9.43**
1.0	8.62	3.95	6.43	**7.29**

(f) AG News: Local test set

p	FedAvg	Local	Fine-tuned	Mixture
0.5	81.93	34.22	78.48	**80.53**
0.7	79.23	48.92	80.37	**81.98**
0.8	75.45	56.21	81.49	**82.66**
0.9	68.47	63.71	**83.44**	82.96
1.0	54.86	72.34	**87.26**	86.51

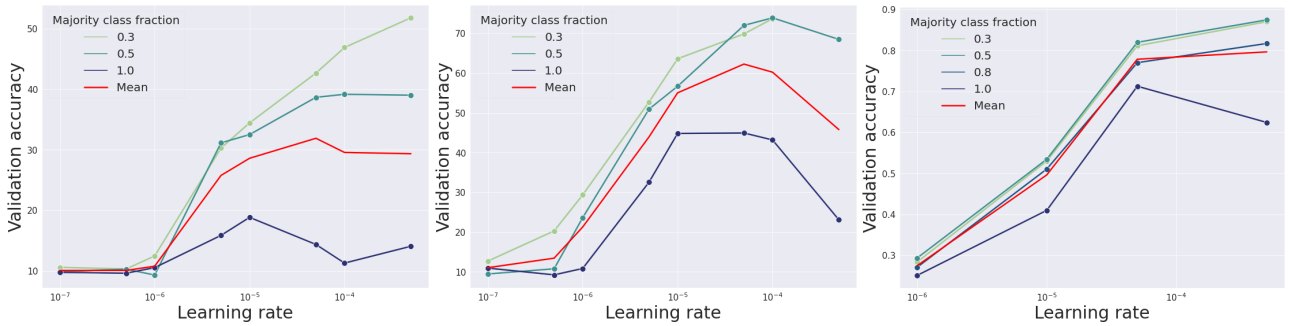
three datasets using different majority class fractions p. The sweep was carried out over learning rates $\eta = \{10^{-7}, 5 \cdot 10^{-7}, \ldots, 10^{-3}, 5 \cdot 10^{-3}\}$. The accuracy was calculated on a balanced validation set. The learning rate of $\eta = 5 \cdot 10^{-5}$ yielded the best validation accuracy for both CIFAR-10 and Fashion-MNIST, and given these results, we use this learning rate for training FEDAVG in all experiments for these two datasets. For the AG News dataset best overall performing learning rate was found to be $\eta = 5 \cdot 10^{-4}$.

The same learning rates of $\eta = 5 \cdot 10^{-5}$ (CIFAR-10 and Fashion-MNIST) and $\eta = 5 \cdot 10^{-4}$ (AG News) was set to train the local baseline models. For the fine-tuned baseline model and the mixture of experts, a lower learning rate was used of $\eta = 10^{-5}$ for CIFAR-10 and Fashion-MNIST and $\eta = 10^{-6}$ for AG News. In the appendix we show training and validation losses for FEDAVG over communication rounds.

In Table 1 we see results for the three datasets for varying majority class fractions p. The leftmost Tables 1a, 1c and 1e show the results for a global (balanced) dataset, which is the same for all clients. The rightmost tables 1b, 1d and 1f show the results on a local (unbalanced) dataset mirroring each clients distribution. In bold we present the best performing specialist. We note here that our proposed mixture of experts is overall the best specialist model in terms of generalization, whereas it performs roughly equally as good as the fine-tuned specialist on a local test set.

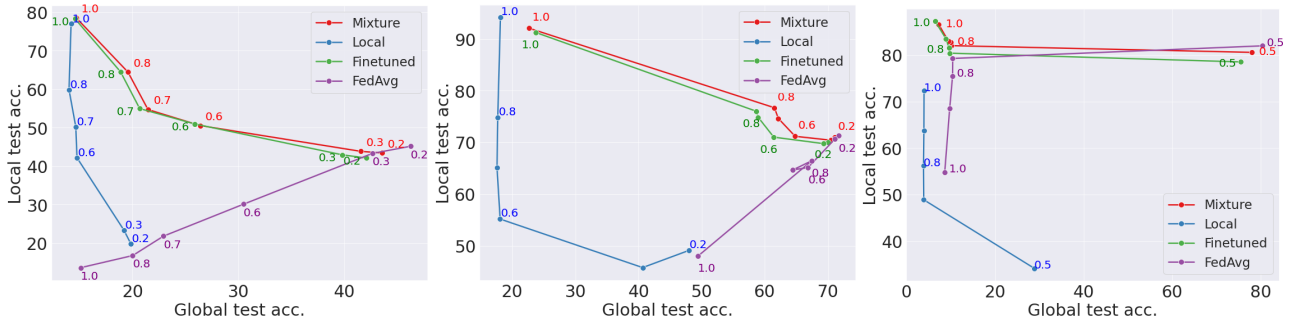
This is further visualized for all datasets in Figure 3. Here the global test accuracy is shown on the x-axis and the local test accuracy is shown on the y-axis for the different baselines. We note that the mixture of expert consistently outperforms the fine-tuned specialist on all three datasets, performing roughly equal on a local test set while consistently outperforming it on the global test set.

In Figure 4 global test accuracies for the fine-tuned baseline and the mixture of experts on all datasets are shown, as a fraction of FEDAVG test accuracy. Here we see that that our proposed method consistently outperforms the fine-tuned baseline in terms of generalization, being the closest to FEDAVG performance in all settings. In Figure 5 similar results for different Dirichlet α values are presented, instead of majority class fractions p.



(a) CIFAR-10 (b) Fashion-MNIST (c) AG News

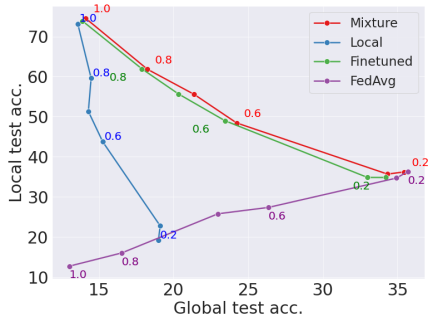
Figure 2. Learning rate vs balanced validation accuracy for FEDAVG on (a) CIFAR-10, (b) Fashion-MNIST and (c) AG News using different majority class fractions p. Reported values are means over four runs.



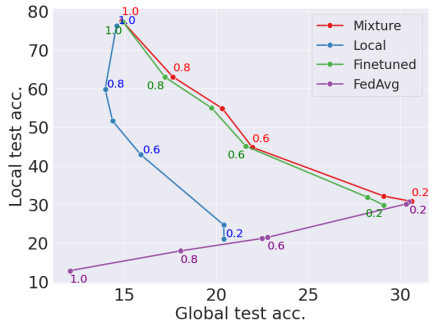
(a) CIFAR-10 (100 clients) (b) Fashion-MNIST (100 clients) (c) AG News (1000 clients)

Figure 3. Accuracy on balanced test data (x-axis) vs unbalanced local test data (y-axis) for the three datasets with opt-out fraction $q = 0$. Different majority class fractions p are shown as colored numbers. Reported values are means over four runs.

Opt-out fractions. Experiments were carried out to test what effect client opt-out has on performance. The results can be seen in Figure 6 for CIFAR-10 over varying majority class fractions p with large opt-out fractions of $q = \{0.9, 0.95\}$, meaning that 90% and 95% of clients, respectively, choose not to participate in the federated learning, but still obtains the global model at the end of training. Similar to the results with no opt-out ($q = 0$), our proposed model outperforms the fine-tuned baseline on the global test set, while performing on par on the local test sets. In Figure 6b, we see that in the iid case of $p = 0.2$ that the mixture not only outperforms the fine-tuned model, but also FEDAVG in both generalization and specialization. This shows that the mixture in this setting is more robust to many clients opting out from federated learning.



(a) Opt out-fraction $q = 0.9$

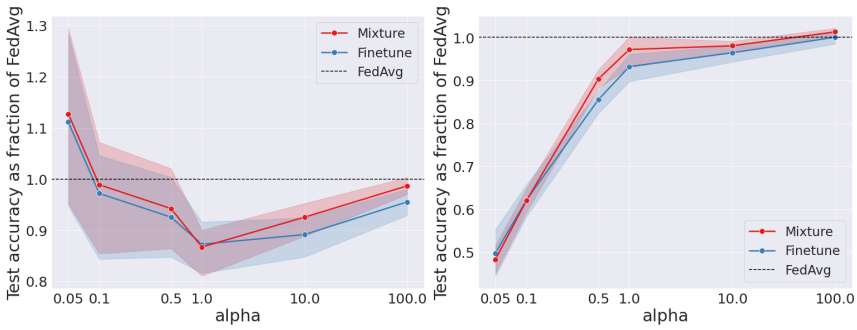


(b) Opt-out fraction $q = 0.95$

Figure 6. Test accuracy on a global test set (x-axis) and local test set (y-axis) for the CIFAR-10 dataset, with two different opt-out fractions q. Majority class fractions are shown in colored numbers.

(a) CIFAR-10 (100 clients) (b) AG News (1000 clients) (c) Fashion-MNIST (100 clients)

Figure 4. Test accuracy with a 95% confidence interval vs majority class fractions for fine-tuned baseline and the mixture on a global (balanced) test set for (a) CIFAR-10 and (b) AG News and (c) Fashion-MNIST, as a fraction of FEDAVG test accuracy. Opt-out fraction $q = 0$. Reported values are means over four runs.



(a) CIFAR-10 (100 clients) (b) Fashion-MNIST (100 clients)

Figure 5. Test accuracy with a 95% confidence interval vs Dirichlet parameter α for fine-tuned baseline and the mixture on a global (balanced) test set for (a) CIFAR-10 and (b) Fashion-MNIST, as a fraction of FEDAVG test accuracy. Opt-out fraction $q = 0$. Reported values are means over four runs.

6. Conclusions

To address the problems of learning a personalized model in a federated setting when the client data is heterogeneous, we have proposed a novel framework for federated mixtures of experts where a global model is combined with a local specialist model. We find that by combining the two expert models we achieve high performance on local client datasets, with minimal loss on generalization as compared to a fine-tuned baseline on two image classification datasets and one text classification dataset in highly non-iid settings, and achieve test accuracies on par with FEDAVG in iid settings.

Our approach is not only an intuitive approach for the generalist vs specialist balance, but also allows for varying level of participation of the different clients in the federation. As such, the framework gives strong privacy guarantees, where clients who do not want to disclose their data are able to opt out and keep their data completely private. The experiments show that our proposed solution is robust to a high opt-out fraction of users, as seen in Figure 3. It thus constitutes a flexible solution for strong privacy guarantees in real-world settings where users might not want to disclose their model to a central server.

The proposed framework is compatible with any gradient-based machine learning model, and can incorporate combinations of these, strengthening the potential of this direction of research, and leveraging the beneficial properties of ensembles of various machine learning models.

In this work we limited our experiments to non-identical distributions in the form of prior probability shift. We hypothesize that our proposed method can handle other distributional shifts, such as covariate or concept shifts as well, and see this as an interesting direction for future work.

References

Arivazhagan, M. G., Aggarwal, V., Singh, A. K., and Choudhary, S. Federated learning with personalization layers. *arXiv preprint arXiv:1912.00818*, 2019.

Bellet, A., Guerraoui, R., Taziki, M., and Tommasi, M. Personalized and private peer-to-peer machine learning. In *International Conference on Artificial Intelligence and Statistics*, pp. 473–481, 2018.

Deng, Y., Kamani, M. M., and Mahdavi, M. Adaptive personalized federated learning. *arXiv preprint arXiv:2003.13461*, 2020.

Fallah, A., Mokhtari, A., and Ozdaglar, A. Personalized federated learning: A meta-learning approach. *arXiv preprint arXiv:2002.07948*, 2020.

Finn, C., Abbeel, P., and Levine, S. Model-agnostic meta-learning for fast adaptation of deep networks. In *ICML*, 2017.

Gulli, A. Ag's corpus of news articles. *URL http://groups. di. unipi. it/˜ gulli/AG_ corpus_of_news_articles. html*, 2004.

Hanzely, F. and Richtárik, P. Federated learning of a mixture of global and local models. *arXiv preprint arXiv:2002.05516*, 2020.

Hard, A., Kiddon, C. M., Ramage, D., Beaufays, F., Eichner, H., Rao, K., Mathews, R., and Augenstein, S. Federated learning for mobile keyboard prediction, 2018. URL https://arxiv.org/abs/1811.03604.

He, C., Avestimehr, S., and Annavaram, M. Group knowledge transfer: Collaborative training of large cnns on the edge. *Advances in Neural Information Processing Systems 33 proceedings (NeurIPS)*, 2020.

Hsieh, K., Phanishayee, A., Mutlu, O., and Gibbons, P. The non-iid data quagmire of decentralized machine learning. In *International Conference on Machine Learning*, pp. 4387–4398. PMLR, 2020.

Hsu, T.-M. H., Qi, H., and Brown, M. Measuring the effects of non-identical data distribution for federated visual classification. *arXiv preprint arXiv:1909.06335*, 2019.

Jacobs, R. A., Jordan, M. I., Nowlan, S. J., and Hinton, G. E. Adaptive mixtures of local experts. *Neural computation*, 3(1):79–87, 1991.

Jeong, E., Oh, S., Kim, H., Park, J., Bennis, M., and Kim, S.-L. Communication-efficient on-device machine learning: Federated distillation and augmentation under non-iid private data. *arXiv preprint arXiv:1811.11479*, 2018.

Jiang, Y., Konečnỳ, J., Rush, K., and Kannan, S. Improving federated learning personalization via model agnostic meta learning. *arXiv preprint arXiv:1909.12488*, 2019.

Johnson, J. M. and Khoshgoftaar, T. M. Survey on deep learning with class imbalance. *Journal of Big Data*, 6(1): 27, 2019.

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. *arXiv preprint arXiv:1912.04977*, 2019.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Konečnỳ, J., McMahan, H. B., Yu, F. X., Richtárik, P., Suresh, A. T., and Bacon, D. Federated learning: Strategies for improving communication efficiency. *arXiv preprint arXiv:1610.05492*, 2016.

Krizhevsky, A., Hinton, G., et al. Learning multiple layers of features from tiny images. 2009.

Lin, T., Kong, L., Stich, S. U., and Jaggi, M. Ensemble distillation for robust model fusion in federated learning. *Advances in Neural Information Processing Systems*, 33, 2020.

Mansour, Y., Mohri, M., and Rostamizadeh, A. Domain adaptation: Learning bounds and algorithms. *arXiv preprint arXiv:0902.3430*, 2009.

McMahan, B., Moore, E., Ramage, D., Hampson, S., and y Arcas, B. A. Communication-efficient learning of deep networks from decentralized data. In *Artificial Intelligence and Statistics*, pp. 1273–1282. PMLR, 2017.

Oquab, M., Bottou, L., Laptev, I., and Sivic, J. Learning and transferring mid-level image representations using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1717–1724, 2014.

Peterson, D., Kanani, P., and Marathe, V. J. Private federated learning with domain adaptation. *arXiv preprint arXiv:1912.06733*, 2019.

Shokri, R. and Shmatikov, V. Privacy-preserving deep learning. In *Proceedings of the 22nd ACM SIGSAC conference on computer and communications security*, pp. 1310–1321, 2015.

Vanhaesebrouck, P., Bellet, A., and Tommasi, M. Decentralized collaborative learning of personalized models over networks. In *Artificial Intelligence and Statistics*, pp. 509–517. PMLR, 2017.

Wang, K., Mathews, R., Kiddon, C., Eichner, H., Beaufays, F., and Ramage, D. Federated evaluation of on-device personalization. *arXiv preprint arXiv:1910.10252*, 2019.

Wang, Z., Song, M., Zhang, Z., Song, Y., Wang, Q., and Qi, H. Beyond inferring class representatives: User-level privacy leakage from federated learning. In *IEEE INFOCOM 2019 - IEEE Conference on Computer Communications*, pp. 2512–2520, 2019.

Xiao, H., Rasul, K., and Vollgraf, R. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms, 2017.

Yurochkin, M., Agarwal, M., Ghosh, S., Greenewald, K., Hoang, N., and Khazaeni, Y. Bayesian nonparametric federated learning of neural networks. In *International Conference on Machine Learning*, pp. 7252–7261. PMLR, 2019.

Zhao, Y., Li, M., Lai, L., Suda, N., Civin, D., and Chandra, V. Federated learning with non-iid data. *arXiv preprint arXiv:1806.00582*, 2018.