FEDERATED LEARNING USING A MIXTURE OF EXPERTS

Edvin Listo Zec, John Martinsson, Olof Mogren, Leon René Sütfeld, Daniel Gillblad
RISE Research Institutes of Sweden
edvin.listo.zec@ri.se

ABSTRACT

Federated learning has received attention for its efficiency and privacy benefits, in settings where data is distributed among devices. Although federated learning shows significant promise as a key approach when data cannot be shared or centralized, current incarnations show limited privacy properties and have shortcomings when applied to common real-world scenarios. One such scenario is heterogeneous data among devices, where data may come from different generating distributions. In this paper, we propose a federated learning framework using a mixture of experts to balance the specialist nature of a locally trained model with the generalist knowledge of a global model in a federated learning setting. Our results show that the mixture of experts model is better suited as a personalized model for devices when data is heterogeneous, outperforming both global and local models. Furthermore, our framework gives strict privacy guarantees, which allows clients to select parts of their data that may be excluded from the federation. The evaluation shows that the proposed solution is robust to the setting where some users require a strict privacy setting and do not disclose their models to a central server at all, opting out from the federation partially or entirely. The proposed framework is general enough to include any kind of machine learning models, and can even use combinations of different kinds.

1 INTRODUCTION

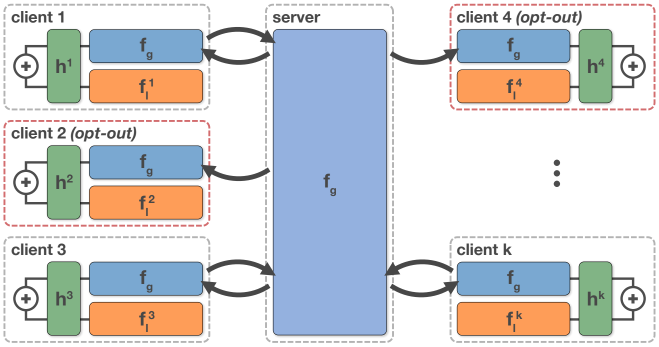

Figure 1: Overview: Federated mixtures of experts using local gating functions.

In many real-world scenarios, data is distributed over a large number of devices, 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 exploiting both the compute power of all participating clients, and to benefit from a large joint training data set. Furthermore, this is beneficial for privacy and data security. For example, 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 extremely sensitive and may have to stay on premise, and a setting where analysis may require distributed and privacy-protecting approaches. In settings with such firm privacy requirements, standard federated learning approaches may not be enough to guarantee the needed privacy.

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

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

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 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 via layer-wise averaging of parameters, weighted by the size of their respective local datasets:

$$w^g_{t+1} \leftarrow \sum_k \frac{n_k}{n} w^k_{t+1}, \tag{2}$$

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.

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. 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.

To address the issue of specialized local models within the federated learning setting, we propose a general framework based on mixtures of experts of local and global models on each client. Local expert models on each client are trained in parallel to the global model, followed by training local gating functions $h^k(x)$ that aggregate the two models' output depending on the input. We show advantages of this approach over fine-tuning the global model on local data in a variety of settings, and analyze the effect that different levels of variation between the local data distributions have on 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 weights communicated to the server (Wang et al., 2019). To this end, in this paper we will work with a stronger notion of privacy. While existing solutions may be private enough for some settings, we will assume that a client that require privacy for some of its data, needs this data to not influence the training of the global model at all. Instead, our framework allows for complete opting out from the federation with all or some of the data at any given 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 evaluation demonstrate the robustness of our learning framework with different levels of skewness 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 with 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., 2016; Bellet et al., 2018). The federated averaging technique (McMahan et al., 2017) has been influential and demonstrated that layer-wise averaging of the weights in neural network models trained separately at the clients is successful in many settings, producing a federated model that demonstrates some 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 (the non-IID setting), which shows that there is a need for personalization within federated learning (Kairouz et al., 2019).

In general, addressing class imbalance with deep learning is still a relatively understudied problem (Johnson & Khoshgoftaar, 2019). A common approach for personalization 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). This approach was proposed for the distributed setting by Wang et al. (2019) who used federated averaging to obtain a generalist model which was later fine-tuned locally on each client, using its 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.

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. 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 $\alpha_i^* = \arg\min_{\alpha_i \in [0,1]} f_i \left(\alpha_i \boldsymbol{v} + (1 - \alpha_i) \boldsymbol{w}\right)$ every communication round. While this weighting scheme will balance the two models, it has no way of adapting to the strengths of the different members of the mix.

Mixture of experts (Jacobs et al., 1991) is the combination of several competing neural networks trained together with a gating network to solve a common task. It was presented as an ensemble method which can be trained end to end using gradient descent. In the current work, we will apply the mixture to leverage the specific strengths of a global model trained with federated averaging, and a local model trained locally on each client.

3 Federated learning using a mixture of experts

In this work, we present a framework for federated learning that builds on federated averaging and mixtures of experts. Our framework includes a personalized model for each client, which is included in a mixture together with a shared globally trained model. The local models never leave the clients, which gives strong privacy properties, while the shared global model is trained using federated averaging, and leverage larger compute and data.

Let f_g be the global model with parameters w_g. We denote the number of clients by k and the local models by f_l^k with parameters w_l^k. The gating function is called 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 model f_l^k is trained using all available data on a client. Third, f_g and f_l^k are trained together with a gating model h^k. In this step, opt-in data may be used to update all three models, while opt-out data may be used only to update f_l^k and h^k. The first two steps may be performed in parallel if allowed by the available resources.

3.1 Privacy guarantees

The proposed framework allows for a strict form of privacy guarantee. Each client may choose an arbitrary part of their data which they consider being too sensitive to use for federated learning, and no information from this data will ever leave the client. The system will still leverage learning from this data by using it to train the local model f_l^k and the gating model h^k. This is a very flexible and useful property. For example, this allows for a user to use the sensitive data in training of the private part, while transforming it using some privatization mechanism and use the censored version to train the federated model.

In general, each client k maintains two different datasets, an opt-out dataset $\mathcal{D}^k_{\mathcal{O}}$ and an opt-in dataset $\mathcal{D}^k_{\mathcal{I}}$. At least one of these has to be non-empty. The local model f^k_l and the gating model h^k will be trained using $\mathcal{D}^k_{\mathcal{O}}$, and the whole mixture (including the global model f_g) will be trained using $\mathcal{D}^k_{\mathcal{I}}$.

3.2 Optimization

Step 1. We train the global model using FedAvg. In other words, globally we optimize

$$\min_{w^g \in \mathbb{R}} \mathcal{L}_{\text{global}}(w^g) \tag{3}$$

where

$$\mathcal{L}_{\text{global}}(w^g) = \frac{1}{n} \sum_{k=1}^{n} \mathbb{E}_{(x,y)\sim\mathcal{D}^k_{\mathcal{I}}} \left[\ell_k(w^g;\ x, y, \hat{y}_g)\right], \tag{4}$$

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. The local models f^k_l are trained only locally, sharing no information between clients, minimizing the the local loss over $w^k_l \in \mathbb{R}$,

$$\mathcal{L}(w^k_l) = \mathbb{E}_{(x,y)\sim\mathcal{D}^k_{\mathcal{O}}} \left[\ell_k(w^k_l;\ x, y, \hat{y}_l)\right] \quad \forall k = 1, \ldots, n \tag{5}$$

where $\hat{y}_l = f^k_l(w^k_l;\ x)$ is the prediction from the local model on the input x, and $\mathcal{D}^k_{\mathcal{O}}$ is the opt-out data distribution for client k.

Step 3. 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^k_l:

$$\hat{y}_{\text{mix}} = h^k(x) f^k_l(x) + \left(1 - h^k(x)\right) f_g(x) \quad \forall k = 1, \ldots, n. \tag{6}$$

In other words, at the end of a communication round, given f^k_l and f_g, we optimize the mixture equation 6 by solving $\min \mathcal{L}_{\text{mix}}$ over w_g, w^k_l, w^k_h, where

$$\mathcal{L}_{\text{mix}}(w_g, w^k_l, w^k_h) = \mathbb{E}_{(x,y)\sim\mathcal{D}^k_{\mathcal{I}}} \left[\ell_k(w_g, w^k_l, w^k_h;\ x, y, \hat{y}_{\text{mix}})\right]. \tag{7}$$

This is done locally for every client $k = 1, \ldots, n$. Here ℓ_k is the loss from predicting $\hat{y}_{\text{mix}}$ for the label y given the input x with the model from equation 6 over the opt-in data distribution $\mathcal{D}^k_{\mathcal{I}}$ of client k.

3.3 Experimental setup

Dataset. Our experiments are carried out on the datasets CIFAR-10 and CIFAR-100 (Krizhevsky et al., 2009). In order to simulate heterogeneous client data, we partition the data into 5 clients for CIFAR-10, and 50 clients for CIFAR-100. The datasets are sampled in such a way that each client yields two majority classes which together form $p\%$ of the client data and the remaining classes form $(1 - p)\%$ of the client data. We perform experiments where we vary p to see what effect the degree of heterogeneity has on performance. In the extreme case $p = 1.0$, each client only has two labels in total. There is no overlap of labels between clients.

Opt-out factor. Some users might want to opt-out from participating to a global model, due to privacy reasons. These users will still receive a 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, illustrated in Figure 1. These clients have all their data in $\mathcal{D}^k_{\mathcal{I}}$, while the rest of the clients have all their data in $\mathcal{D}^k_{\mathcal{O}}$. $q = 0$ means all clients are opt-in and participating. We perform experiments varying q, to see how robust our algorithm is to different levels of client participation.

Models. In our setup, both the local model f_l and the global model f_g are CNNs with the same architecture. However, they are not constrained to be the same model and could be implemented any two differentiable models. The CNN has two convolutional layers with a kernel size of 5, and two linear layers. All layers have ReLU activations. The gating function h has the same architecture as f_g and f_l, but with a sigmoid activation in the last layer.

Baselines. We use three different models as baselines. First, the locally trained model f_l^k for each client. Second, FedAvg. Third, the final model output from FedAvg fine-tuned for each client on its own local data. We train f_l^k, the fine-tuned model and the mixture using early stopping for 100 epochs, monitoring validation loss on each client. We train f_g using FedAvg with 45 communication rounds and 3 local epochs in all experiments. Further, we use Adam (Kingma & Ba, 2014) to optimize all models, with a learning rate of 0.0001.

Evaluation. For evaluation we have a held-out validation set for each client. For both CIFAR-10 and CIFAR-100 we have $n = 400$ data points for evaluation per client, sampled with the same majority class fraction p. We report an average accuracy over all clients.

4 Results

For the sake of reproducibility, all code will be made available.

In Table 1 we report accuracies and standard deviations on CIFAR-10 for all models when data is highly non-iid, i.e. for $p = \{0.8, 0.9, 1.0\}$. In Figures 2 and 3 we report average accuracies over three runs for all majority fractions p on the datasets CIFAR100 and CIFAR-10, respectively.

In Figure 2 we see that the mixture model outperforms all other models on CIFAR-100 for all p. Figure 3 shows that the mixture outperforms both the locally trained model and the fine-tuned model on highly skewed data, i.e. for $p > 0.5$. In both figures we also see that FedAvg is degrading in performance as majority class fraction p increases, due to client distributions becoming too heterogeneous. In Figure 2 we also see that the fine-tuned baseline performs worse than the locally trained model when FedAvg degrades in performance.

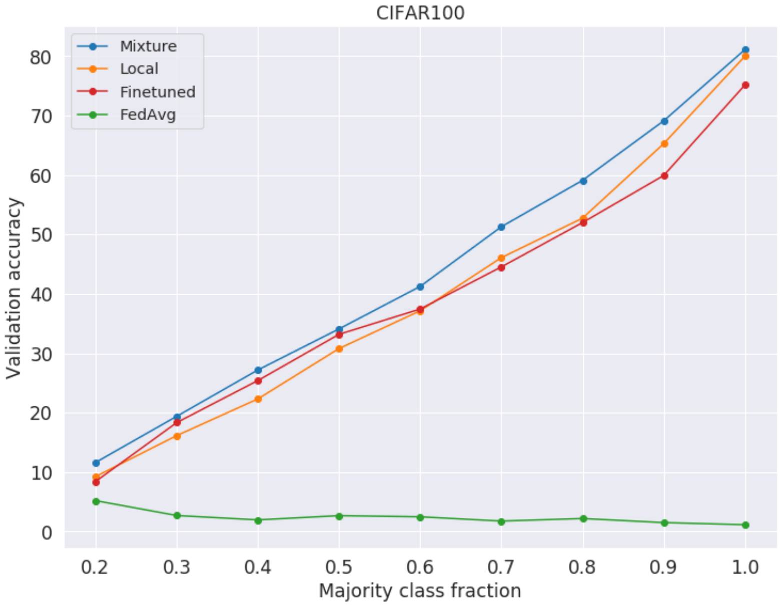

Figure 2: Accuracy on unbalanced local validation data for CIFAR-100 for different majority class fractions p. Opt-out factor $q = 0.0$. All accuracies plotted are means over three runs.

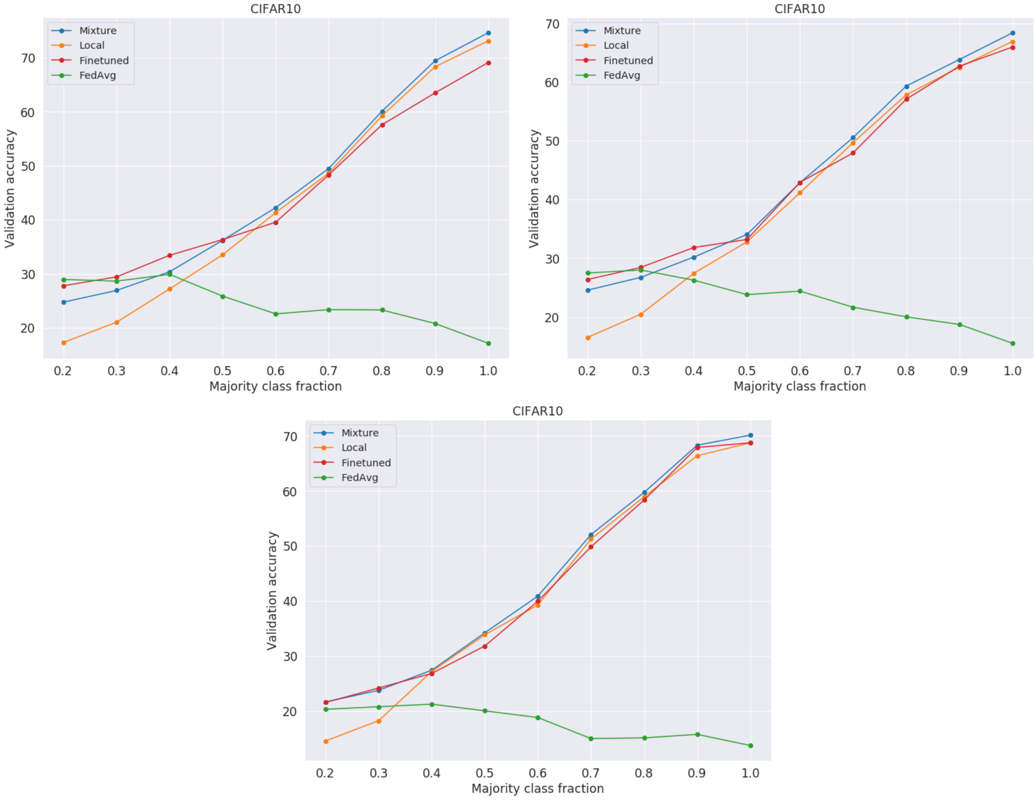

Figure 3: Accuracy on unbalanced local validation data for CIFAR-10 for different majority class fractions p. Top left plot shows opt-out factor $q = 0.0$, meaning no clients opt-out from federation. Top right plot shows opt-out factor $q = 0.5$, meaning 50% of clients opt-out from federation. Bottom plot shows opt-out factor $q = 0.9$, meaning 90% of clients opt-out from federation. All accuracies plotted are means over three runs.

p	FedAvg	Local	Fine-tuned	Mixture
1.0	17.13 ± 1.22	73.13 ± 2.00	69.12 ± 1.78	$\mathbf{74.62 \pm 3.35}$
0.9	20.79 ± 0.81	68.32 ± 1.51	63.49 ± 0.96	$\mathbf{69.44 \pm 1.44}$
0.8	23.29 ± 2.68	59.25 ± 3.02	57.59 ± 2.59	$\mathbf{60.10 \pm 1.54}$

Table 1: Mean accuracy on unbalanced validation set for highly non-iid majority class fractions p and models. Means and standard deviations reported are over three runs. Opt-out fraction $q = 0.0$.

	CIFAR-10	CIFAR-100
Model	CNN	CNN
No. of clients	5	50
Training data size per client	100	100
Validation data size per client	400	400

Table 2: Experimental set-up summary.

5 Discussion

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 local specialist models from every client. We find that with skewed non-IID data on the clients, our approach outperforms all other baselines, including federated averaging, locally trained models, and models trained first with federated averaging and then fine-tuned on each local client. The experimental evaluation for CIFAR-10 shows that for more heterogeneous data, $p > 0.5$, our approach outperforms all other methods, including the strong fine-tuning baseline (see Figure 3). For CIFAR-100, the proposed framework outperforms all other methods, regardless of the level of skewness (see Figure 2). In this setting, a large part of the training data for each client comes from a very limited set of the available classes (two out of 100), and very few training examples will be available from the minority classes. This is a crucial result: the proposed framework is very robust to extremely skewed training data.

The framework also gives strong privacy guarantees, and the experiments show that our proposed solution is robust to a high opt-out fraction of users, in fact, for all examined fractions of opt-out users q, we consistently outperform the baselines for $p > 0.5$ (see Figure 3).

6 Conclusions

In this work, we have presented a framework for federated learning that builds on mixtures of experts. This framework allows us to learn a model that has a balance between the generalist nature of the global federated model and the specialist nature of the local client models.

Our approach is not only an intuitive approach for the generalist vs specialist balance, but also allows for varying participation of the different clients in the federation: clients may either opt-in entirely, keep a part of their data entirely private (training only its local model with that part, and the rest for the federated model), or opt-out entirely (by training only a local model with all its local data). This gives a flexible solution for strong privacy guarantees in real settings.

The proposed framework is able to include any kind of machine learning models, and can even incorporate combinations of them, further strengthening the potential of this direction of research, and leveraging the beneficial properties of ensembles of various machine learning models.

In the experimental evaluation, we have demonstrated that our solution leads to state-of-the-art results in two different benchmark datasets when data is skewed, and when parts of the clients in the federation opts out from the training.

References

Manoj Ghuhan Arivazhagan, Vinay Aggarwal, Aaditya Kumar Singh, and Sunav Choudhary. Federated learning with personalization layers. *arXiv preprint arXiv:1912.00818*, 2019.

Aurélien Bellet, Rachid Guerraoui, Mahsa Taziki, and Marc Tommasi. Personalized and private peer-to-peer machine learning. In *International Conference on Artificial Intelligence and Statistics*, pp. 473–481, 2018.

Yuyang Deng, Mohammad Mahdi Kamani, and Mehrdad Mahdavi. Adaptive personalized federated learning. *arXiv preprint arXiv:2003.13461*, 2020.

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

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *ICML*, 2017.

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

Andrew Hard, Chloé M Kiddon, Daniel Ramage, Francoise Beaufays, Hubert Eichner, Kanishka Rao, Rajiv Mathews, and Sean Augenstein. Federated learning for mobile keyboard prediction, 2018. URL `https://arxiv.org/abs/1811.03604`.

Robert A Jacobs, Michael I Jordan, Steven J Nowlan, and Geoffrey E Hinton. Adaptive mixtures of local experts. *Neural computation*, 3(1):79–87, 1991.

Yihan Jiang, Jakub Konečnỳ, Keith Rush, and Sreeram Kannan. Improving federated learning personalization via model agnostic meta learning. *arXiv preprint arXiv:1909.12488*, 2019.

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

Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Keith Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. Advances and open problems in federated learning. *arXiv preprint arXiv:1912.04977*, 2019.

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

Jakub Konečnỳ, H Brendan McMahan, Felix X Yu, Peter Richtárik, Ananda Theertha Suresh, and Dave Bacon. Federated learning: Strategies for improving communication efficiency. *arXiv preprint arXiv:1610.05492*, 2016.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Yishay Mansour, Mehryar Mohri, and Afshin Rostamizadeh. Domain adaptation: Learning bounds and algorithms. *arXiv preprint arXiv:0902.3430*, 2009.

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*, pp. 1273–1282. PMLR, 2017.

Maxime Oquab, Leon Bottou, Ivan Laptev, and Josef Sivic. 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.

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

Paul Vanhaesebrouck, Aurélien Bellet, and Marc Tommasi. Decentralized collaborative learning of personalized models over networks. *arXiv preprint arXiv:1610.05202*, 2016.

Kangkang Wang, Rajiv Mathews, Chloé Kiddon, Hubert Eichner, Françoise Beaufays, and Daniel Ramage. Federated evaluation of on-device personalization. *arXiv preprint arXiv:1910.10252*, 2019.

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