Efficient and Light-Weight Federated Learning via Asynchronous Distributed Dropout

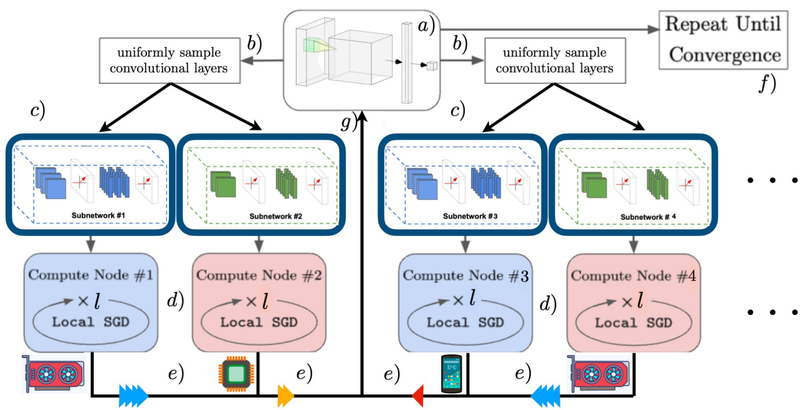
Chen Dun, Mirian Del Carmen Hipolito Garcia, Dimitrios Dimitriadis, Christopher Jermaine, Anastasios Kyrillidis

Background and Motivation

Edge devices in Federated Learning are extremely heterogeneous in terms of compute power, device memory and communication bandwidths. Directly applying common synchronized FL algorithms results often in a ``stragglers'' effect. On other hand, asynchronous FL algorithms might suffer from drop in final accuracy due to gradient staleness and gradient overfitting, resulting from inconsistent update schedule between fast and slow clients. As in most asynchronous algorithms one still need to broadcast full model to all devices, following a data parallel distributed protocol, regardless of device heterogeneity. This inspire us to ask a key question: ***"Can we select sub-models out of the global model and send these to each device, taking into account the device heterogeneity?"***

Method

We propose (Hetero) AsyncDrop, a novel asynchronous FL framework with smart (i.e., informed/structured) dropout that achieves better performance compared to state of the art asynchronous methodologies, while resulting in less communication and training time costs. The key idea revolves around creating sub-models out of the global model and distributing their training to workers, that take into account the device heterogeneity.



Algorithm 3 `Hetero AsyncDrop` for Asynchronous FL

Parameters: T iters, S clients, l local iters., $\mathbf{W}$ as current global model, $\mathbf{W}_i$ as local model for i-th client, η_g as global LR, $v(\cdot)$ weight score function, $\psi(i)$ computes the computation capacity of i-th worker, $\varphi(\mathbf{W}, \psi(i), v(\cdot))$ is the `Smart Dropout` function that creates the mask, based on worker capacity ψ and score v, $\alpha \in (0, 1)$.

$\mathbf{W} \leftarrow$ randomly initialized global model.
`//On each client` i `asynchronously:`
for $t = 0, \dots, T-1$ **do**
  `//For` i`-th fastest worker,` $\varphi(\cdot)$ `drops weights with` i`-th largest` $v(\cdot)$ `score`
  Generate mask $\mathbf{M}_{i,t} = \varphi(\mathbf{W}_t, \psi(i), v(\cdot))$
  $\mathbf{W}_{i,t} \leftarrow \mathbf{W}_t \odot \mathbf{M}_{i,t}$
  `//Train` $\mathbf{W}_{i,t}$ `for` l `iters. via SGD`
  for $j = 1, \dots, l$ **do**
    $\mathbf{W}_{i,t} \leftarrow \mathbf{W}_{i,t} - \eta_g \frac{\partial \mathcal{L}}{\mathbf{W}_{i,t}}$
  end for
  if i-th client is fastest **then**
    Update η_g
  end if
  `//Write local to global model`
  $\mathbf{W}_{t+1} \leftarrow \mathbf{W}_t \odot (\mathbf{M}_{i,t})^c + ((1-\alpha) \cdot \mathbf{W}_t + \alpha \cdot \mathbf{W}_{i,t}) \odot \mathbf{M}_{i,t}$
  `//Update score` q
  $v(\mathbf{W}_{t+1}^j) = \left\| \mathbf{W}_{t+1}^j - \mathbf{W}_0^j \right\|_1, \forall\, j \in \mathcal{J}$
end for

Theoratical Analysis

Theorem 3.1 *Let $f(\cdot,\cdot)$ be a one-hidden-layer CNN with the second layer weight fixed. Let $\mathbf{u}_t$ abstractly represent the output of the model after t iterations, over the random selection of the masks. Let ξ denote the dropout rate ($\xi = 1$ dictates that all neurons are selected), and denote $\theta = 1 - (1-\xi)^S$ the probability that a neuron is active in at least one subnetwork. Assume the number of hidden neurons satisfies $m = \Omega\left(\max\{\frac{n^4 K^2}{\lambda_0^4 \delta^2} \max\{n, d\}, \frac{n}{\lambda_0}\}\right)$ and the step size satisfies $\eta = O\left(\frac{\lambda_0}{n^2}\right)$. Let κ be a proper initialization scaling factor, and it is considered constant. We use λ_0 to denote the smallest eigenvalue of the Neural Tangent Kernel matrix. Let Assumptions 1 and 2 be satisfied. Then, the following convergence rate guarantee is proved to be satisfied:*

$$\mathbb{E}_{\mathbf{M}_t}\left[\|\mathbf{u}_{t+1} - \mathbf{y}\|_2^2\right] \leq \left(1 - \frac{\theta\eta\lambda_0}{4}\right)^t \|\mathbf{u}_0 - \mathbf{y}\|_2^2 + O\left(\frac{\theta\eta\lambda_0^3\xi^2\kappa^2E^2}{n^2} + \frac{\xi^2(1-\xi)^2\theta\eta n^3\kappa^2 d}{m\lambda_0} + \frac{\eta^2\theta^2 n\kappa^2\lambda_0\xi^4E^2}{m^4} + \frac{\xi^2(1-\xi)^2\theta^2\eta^2n^2\kappa^2 d}{m^3\lambda_0} + \frac{\xi^2(1-\xi)^2\theta^2\eta^2\kappa^2\lambda_0E^2}{m^3} + \frac{\xi^2(1-\xi)^2\theta^2\eta^2n^2\kappa^2 d}{m^2\lambda_0} + \frac{n\kappa^2\left(\theta - \xi^2\right)}{S}\right)$$

Experiment Results

(Hetero) AsyncDrop is tested on CNN, MLP and LSTM on CIFAR10, CIFAR100, FMNIST and IMDB Sentiment with extreme data and device heterogeneity. In all cases,(Hetero) AsyncDrop shows significant improvement in final accuracy, total communication cost and training time cost.

Table 2: Test accuracy of asynchronous FL baselines vs. (`Hetero`) `AsyncDrop` using a ResNet architecture on non-i.i.d CIFAR10, CIFAR100 and FMNIST data over > 100 clients. We report the time and communication overhead to reach a certain target accuracy: "Time for XX% Accuracy" denotes the second lowest test accuracy among all baselines as the target accuracy. Teal colored text indicates favorable performance; red colored text indicates high variance in performance.

CIFAR10	Max. Test Accuracy	Time for 35% Accuracy	Time Overhead	Comm. Overhead
Async. FedAvg	$45.79 \pm$ **7.9**	2105.5s	+33.34%	+15.56%
Async. Fed-Weighted-Avg	$46.51 \pm$ **6.8**	2105.5s	+33.34%	+15.56%
Async. FedProx	43.97 ± 1.35	2296.9s	+45.46%	+26.06%
Async. FjORD	23.14 ± 0.90	N/A	N/A	N/A
FedBuff	$35.81 \pm$ **11.83**	3012.9s	+89.98%	+35.75%
`Hetero AsyncDrop`	**50.67** ± 1.75	**1579.1s**	**[Best]**	**[Best]**
`AsyncDrop`	$48.98 \pm$ **3.87**	2009.7s	+27.27 %	+27.27%

CIFAR100	Max. Test Accuracy	Time for 32% Accuracy	Time Overhead	Comm. Overhead
Async. FedAvg	32.47 ± 1.89	3062.5s	+33.38%	+15.56%
Async. Fed-Weighted-Avg	32.98 ± 1.71	3062.5s	+33.38%	+15.56%
Async. FedProx	35.75 ± 0.61	3062.5s	+33.38%	+15.56%
Async. FjORD	12.07 ± 0.83	N/A	N/A	N/A
FedBuff	**41.91** $\pm$ **3.80**	4250.1s	+85.03%	+42.22%
`Hetero AsyncDrop`	37.26 ± 0.93	**2296.8s**	**[Best]**	**[Best]**
`AsyncDrop`	35.93 ± 0.93	**2296.8s**	**[Best]**	**[Best]**

FMNIST	Max. Test Accuracy	Time for 57% Accuracy	Time Overhead	Comm. Overhead
Async. FedAvg	$62.47 \pm$ **8.20**	787.5s	+33.33%	+25.00%
Async. Fed-Weighted-Avg	$59.30 \pm$ **10.44**	1181.3s	+100%	+87.50%
Async. FedProx	$59.91 \pm$ **7.32**	1050.0s	+77.78%	+66.67%
Async FjORD	$21.98 \pm$ **9.55**	N/A	N/A	N/A
FedBuff	$57.03 \pm$ **11.37**	1845.0s	+212.38%	+150.20%
`Hetero AsyncDrop`	**66.89** $\pm$ **5.36**	**590.6s**	**[Best]**	**[Best]**
`AsyncDrop`	$60.02 \pm$ **10.38**	787.5s	+33.33%	+33.33%