FedSpeed: Larger Local Interval, Less Communication Round, and Higher Generalization Accuracy

Yan Sun
The University of Sydney
ysun9899@uni.Sydney.edu.au

Li Shen
JD Explore Academy
mathshenli@gmail.com

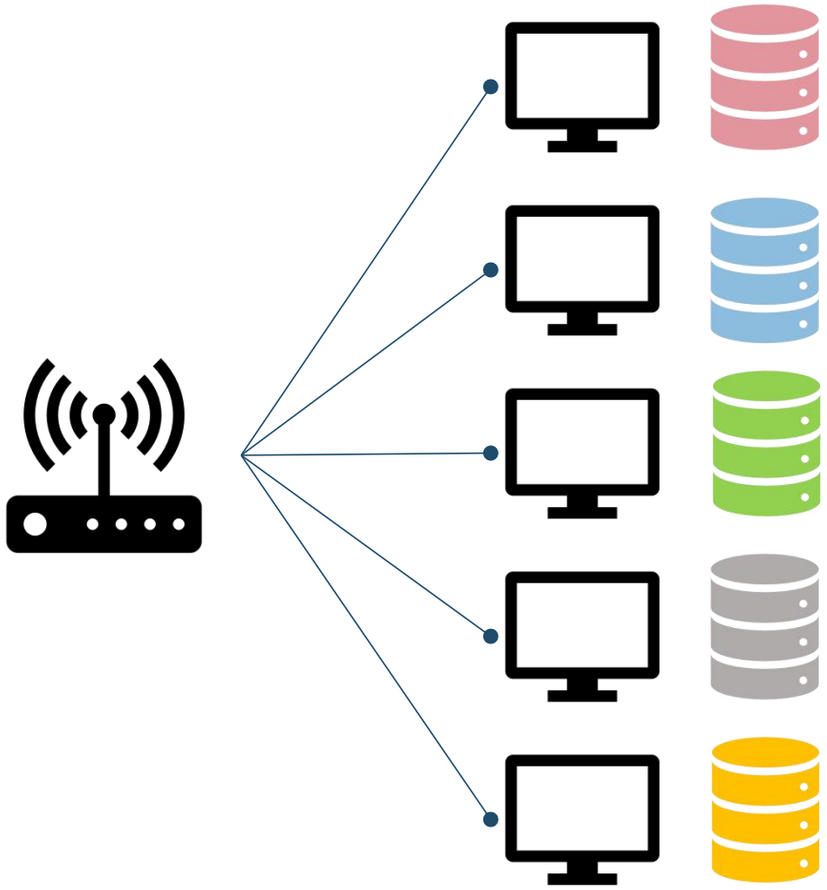
Tiansheng Huang
Georgia Institute of Technology
tianshenghuangscut@gmail.com

Liang Ding
JD Explore Academy
liangding.liam@gmail.com

Dacheng Tao
The University of Sydney
dacheng.tao@gmail.com

Section 01 Federated Learning

Preliminaries – Federated Learning



- Privacy Protection
- Distributed Framework
- Data Shift of Isolated Sources
- Local Limited Calculations
- Communication Bottleneck

Preliminaries – Problem Setup

$$\min\ f(w) = \Sigma_i\, p_i f_i(w),$$

$$\min\ f(w) = \Sigma_i\, p_i f_i(w_i), \quad s.t.\ w_i = w.$$

Where $f_i(w_i) = E_{\{x_i \sim D_i\}} f(w_i; x_i)$ is the ERM loss.

Two main difficulties:

1. Equality constraints : x_i obeys different distribution D_i.
2. Resource constraints: limited capacity for computation and communication.

Preliminaries – FedAvg

Algorithm 1: FedAvg Algorithm

Input: global model w, local model w_i, communication round T, local interval K.
Output: The global model w^T.

Initialize states: initialize $w = w^0$.
for $t = 0, 1, ..., T-1$ **do**
  randomly select the active clients set $[n]$ from $[m]$
  for $i \in [n]$ ***in parallel*** **do**
    send the w^t to the active clients as $w_{i,0}^t$
    for $k = 0, 1, ..., K-1$ **do**
      compute the stochastic gradient $g_{i,k}^t$ at $w_{i,k}^t$
      $w_{i,k+1}^t = w_{i,k}^t - \eta g_{i,k}^t$
    send the $w_i^t = w_{i,K}^t$ to the global server
  $w^{t+1} = \frac{1}{n} \sum_{i \in [n]} w_i^t$
return w^T

$[n]$: *selected active clients.*

$[m]$: *total clients.*

Section 02 FedSpeed

FedSpeed – Perturbation

$$\min\ L_i(w) = f_i(w) + a_i\|\nabla f_i(w)\|^2,$$

$$\begin{aligned}
&\nabla L_i(w) \\
&= \nabla f_i(w) + 2a_i\nabla^2 f_i(w)\cdot\nabla f_i(w) \\
&\approx \nabla f_i(w) + \frac{2a_i\left(\nabla f_i\left(w + r\nabla f_i(w)\right) - \nabla f_i(w)\right)}{r} + O(r^2\|\nabla f_i(w)\|^2) \\
&\approx \frac{(1-2a_i)}{r}\nabla f_i(w) + \frac{2a_i}{r}\nabla f_i(w + r\nabla f_i(w))
\end{aligned}$$

Vanilla Stochastic Gradient

Perturbation Gradient

[1] Penalizing Gradient Norm for Efficiently Improving Generalization in Deep Learning

FedSpeed – ADMM

$$\min\ L(w) = \Sigma_i\, p_i L_i(w_i), \quad s.t.\ w_i = w.$$

Lagrangian

$$\mathbf{L}(w, w_i) = \Sigma_i \{\, p_i L_i(w_i) + \langle \lambda_i, w_i - w \rangle + \frac{\beta}{2} \|w_i - w\|^2 \}$$

(1)Minimize $\mathbf{L}(w_i)$.

(2)Update dual $\lambda_i = \lambda_i + \frac{1}{\beta}(w_i - w)$

(3)Minimize $\mathbf{L}(w)$.

FedSpeed – Algorithm

Algorithm 1 FedSpeed Algorithm Framework

Input: model parameters $\mathbf{x}^0$, total communication rounds T, local gradient controller $\hat{\mathbf{g}}_i^{-1} = 0$, penalized weight λ.

Output: model parameters $\mathbf{x}^T$.

for $t = 0, 1, 2, \cdots, T-1$ **do**
  select active clients-set $\mathcal{S}^t$ at round t
  for client $i \in \mathcal{S}^t$ parallel **do**
    communicate $\mathbf{x}^t$ to local client i and set $\mathbf{x}_{i,0}^t = \mathbf{x}^t$
    for $k = 0, 1, 2, \cdots, K-1$ **do**
      sample a minibatch $\varepsilon_{i,k}^t$ and do
      compute unbiased stochastic gradient: $\mathbf{g}_{i,k,1}^t = \nabla F_i(\mathbf{x}_{i,k}^t; \varepsilon_{i,k}^t)$
      update the extra step: $\breve{\mathbf{x}}_{i,k}^t = \mathbf{x}_{i,k}^t + \rho \mathbf{g}_{i,k,1}^t$
      compute unbiased stochastic gradient: $\mathbf{g}_{i,k,2}^t = \nabla F_i(\breve{\mathbf{x}}_{i,k}^t; \varepsilon_{i,k}^t)$
      compute quasi-gradient: $\tilde{\mathbf{g}}_{i,k}^t = (1-\alpha)\mathbf{g}_{i,k,1}^t + \alpha \mathbf{g}_{i,k,2}^t$
      update the gradient descent step: $\mathbf{x}_{i,k+1}^t = \mathbf{x}_{i,k}^t - \eta_l \left(\tilde{\mathbf{g}}_{i,k}^t - \hat{\mathbf{g}}_i^{t-1} + \frac{1}{\lambda}(\mathbf{x}_{i,k}^t - \mathbf{x}^t) \right)$
    end for
    $\hat{\mathbf{g}}_i^t = \hat{\mathbf{g}}_i^{t-1} - \frac{1}{\lambda}(\mathbf{x}_{i,K}^t - \mathbf{x}^t)$
    communicate $\hat{\mathbf{x}}_i^t = \mathbf{x}_{i,K}^t - \lambda \hat{\mathbf{g}}_i^t$ to the global server
  end for
  $\mathbf{x}^{t+1} = \frac{1}{S} \sum_{i \in \mathcal{S}^t} \hat{\mathbf{x}}_i^t$
end for

Section 03 Convergence

FedSpeed – Convergence

Assumption 1. For the non-convex function f_i, it is L-smooth when:

$$\|\nabla f_i(x) - \nabla f_i(y)\| \leq L\|x - y\|.$$

Assumption 2. Stochastic gradients are zero-mean and bounded-variance estimation:

$$E\left[g_{i,k}^t - \nabla f_i\left(w_{i,k}^t\right)\right] = 0, \quad \left\|g_{i,k}^t - \nabla f_i\left(w_{i,k}^t\right)\right\| \leq \sigma_l^2.$$

Assumption 3. Heterogeneity is bounded:

$$E\|\nabla f_i(w) - \nabla f(w)\| \leq \sigma_g^2$$

FedSpeed – Convergence

Theorem 4.4 *Under the Assumptions 4.1-4.3 when the perturbation learning rate satisfies* $\rho \leq \frac{1}{\sqrt{6}\alpha L}$, *and the local learning rate satisfies* $\eta_l \leq \min\{\frac{1}{32\sqrt{3K}L}, 2\lambda\}$, *and the local interval satisfies* $K \geq \lambda/\eta_l$, *let* $\kappa = \frac{1}{2} - 3\alpha^2 L^2 \rho^2 - 1536\eta_l^2 L^2 K$ *is a positive constant with selecting the proper* η_l *and* ρ, *the auxiliary sequence* $\mathbf{z}^t$ *in Equation (5) generated by executing the Algorithm 1 satisfies:*

$$\frac{1}{T}\sum_{t=1}^{T-1}\mathbb{E}\|\nabla F(\mathbf{z}^t)\|^2 \leq \frac{2(F(\mathbf{z}^1)-F^*)}{\lambda\kappa T} + \frac{64\eta_l L^2 K}{\kappa m T}\sum_{i\in[m]}\mathbb{E}\|\hat{\mathbf{g}}_i^0\|^2 + \frac{32\lambda^2 L^2}{\kappa T}\mathbb{E}_t\|\frac{1}{m}\sum_{i\in[m]}\hat{\mathbf{g}}_i^0\|^2 + \Phi, \tag{6}$$

where F *is a non-convex objective function* F^* *is the optimal of* F. *The term* Φ *is:*

$$\Phi = \frac{1}{\kappa}\left(32\lambda\eta_l^2 L^2 K(16\sigma_g^2 + \sigma_l^2) + \lambda\alpha^2 L^2\rho^2(3\sigma_g^2 + \sigma_l^2)\right), \tag{7}$$

Section 04 Experiments

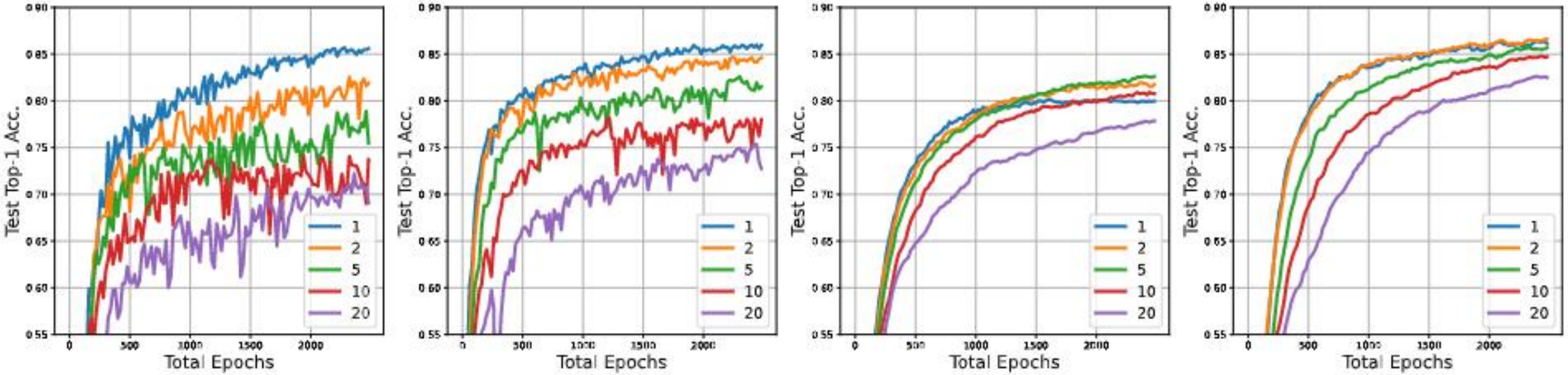
FedSpeed – Experiments

Method	CIFAR-10		CIFAR-100		TinyImagenet	
	IID.	DIR.	IID.	DIR.	IID.	DIR.
FedAvg	77.01	75.21	40.68	39.33	33.58	32.71
FedProx	77.68	75.97	41.29	39.69	33.71	32.78
FedAdam	82.92	80.55	51.65	49.29	40.85	39.71
SCAFFOLD	80.11	77.71	47.38	46.33	38.03	37.54
FedCM	84.20	83.48	52.35	50.98	41.90	41.67
FedDyn	83.36	80.57	46.18	46.60	34.69	33.92
FedADMM	81.29	79.71	45.51	46.65	36.03	33.83
FedSpeed	**85.80**	**84.79**	**53.93**	**52.88**	**43.38**	**42.75**

Experiment Setup:

2%-500 clients, Dirichlet-0.6 split, 2 local epochs with batchsize=20 (25 iteration per round)

FedSpeed – Experiments



(a) FedAdam. (b) FedCM. (c) SCAFFOLD. (d) **FedSpeed.**

Experiment Setup:

10%-100 clients, Dirichlet-0.6 split, batchsize=50

of different local epochs (K*T=constant)

FedSpeed – Experiments

Table 2: Performance of different ρ_0 with $\alpha = 1$.

ρ_0	0	0.01	0.05	**0.1**	0.2
Acc.	83.97	84.6	85.38	**85.72**	84.35

Table 8: Performance of different α with $\rho_0 = 0.1$.

α	0	0.5	0.75	0.875	**0.9375**	1.0
Acc.	83.97	84.36	84.91	85.46	**85.74**	85.72

Experiment Setup:

10%-100 clients, Dirichlet-0.6 split, batchsize=50 of 5 local epochs

FedSpeed – Summary

Main difficulties in the FL(horizontal):

(1)Local consistency (Equality constraint $w = w_i$).

(2)Local over-fitting on the small dataset.

(3)Computational limitation (local devices are poor).

(4)Communication limitation (large scale).

THANK YOU