Federated Optimization in Heterogeneous Networks

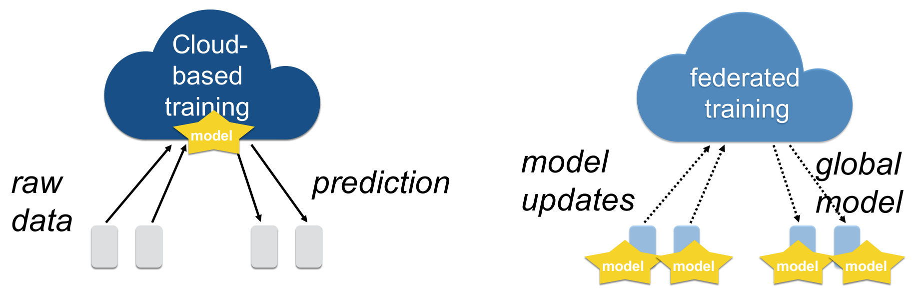
MLSys 2020

Tian Li (CMU), Anit Kumar Sahu (BCAI), Manzil Zaheer (Google Research), Maziar Sanjabi (Facebook AI), Ameet Talwalkar (CMU & Determined AI), Virginia Smith (CMU)

Motivation

Federated learning: privacy-preserving machine learning training in heterogeneous, (potentially) massive networks

Applications: voice recognition/face detection on mobile phones, predictive maintenance, personalized healthcare on wearable devices, applications in smart homes, etc.



Two of the major challenges

Systems heterogeneity

- *Significant variability in terms of systems characteristics on each device in the network (hardware, network, power, etc)*
- Current methods do not allow devices to perform variable amounts of local work

Statistical heterogeneity

- *Non-identically distributed data across the network*
- Lack convergence guarantees and may diverge in practice

Key Ideas

Key idea: Dropping stragglers or naively incorporating partial updates from stragglers implicitly increase statistical heterogeneity

Method: Simple algorithmic modifications to current state-of-the-art method (adding a proximal term to the local subproblem while tolerating partial updates)

Contributions

- (Theoretically) Provide convergence guarantees (rates as functions of statistical and systems heterogeneity)
- (Practically) Allow for more robust convergence (improved absolute accuracy by 22% in highly heterogeneous environments)

FedProx: a Framework for Federated Optimization

Global objective: $\min_{w} f(w) = \mathbb{E}_k\left[F_k(w)\right]$ **Local objective on device k:** $\min_{W_k} F_k(W_k, X_k)$

Idea 1: Allow for **partial** work to be performed on local devices based on systems constraints

Idea 2: At each round, each selected device solves a ***modified*** local subproblem:

$$\min_{W_k} F_k(W_k, X_k) + \frac{\mu}{2}\left\|W_k - W^t\right\|^2$$

A proximal term

- **Generalization of the popular method FedAvg** (FedAvg + allowing for variable local work + proximal term = FedProx)
- **General:** Can use any local solver; theory covers both convex and non-convex losses

The proximal term (1) safely incorporates noisy updates from variable local work; (2) explicitly limits the impact of local updates; (3) makes the method more amenable to theoretical

Proposed FedProx method

Until convergence:

1. Server samples devices, and sends the current global model to all chosen devices
2. Each device solves the following subproblem by performing variable local updates based on the underlying systems constraints
$$\min_{W_k} F_k(W_k, X_k) + \frac{\mu}{2}\left\|W_k - W^t\right\|^2$$
3. Server aggregates local updates and forms a new global model

Convergence Analysis

Characterize statistical heterogeneity: B-dissimilarity $B(w) = \sqrt{\frac{\mathbb{E}_k\left[\|\nabla F_k(w)\|^2\right]}{\|\nabla f(w)\|^2}}$

B quantifies statistical heterogeneity

Assumptions

Assumption 1: Bounded Dissimilarity

Assumption 2: Modified Local subproblem is convex & smooth

Assumption 3: Each local subproblem is solved inexactly to some optimality

introduce γ_k^t-inexactness to capture **systems heterogeneity**

[Theorem] Obtain suboptimality ε, after T iterations, with:

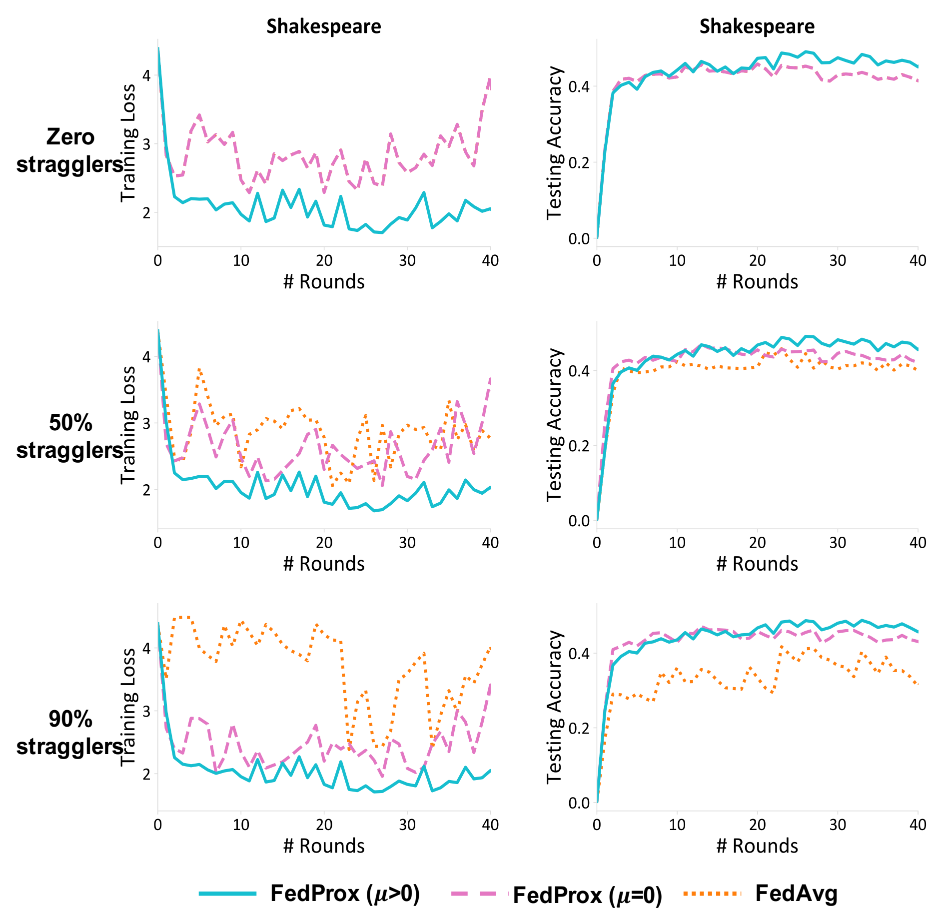
$$T = O\left(\frac{f(w^0) - f^*}{\rho\varepsilon}\right)$$

ρ: related to μ, B, γ_k^t

- **Rate is general**
- Covers both convex, and non-convex loss functions
- Independent of the local solver
- Agnostic of the sampling method
- **The same asymptotic convergence guarantee as SGD**

Evaluation

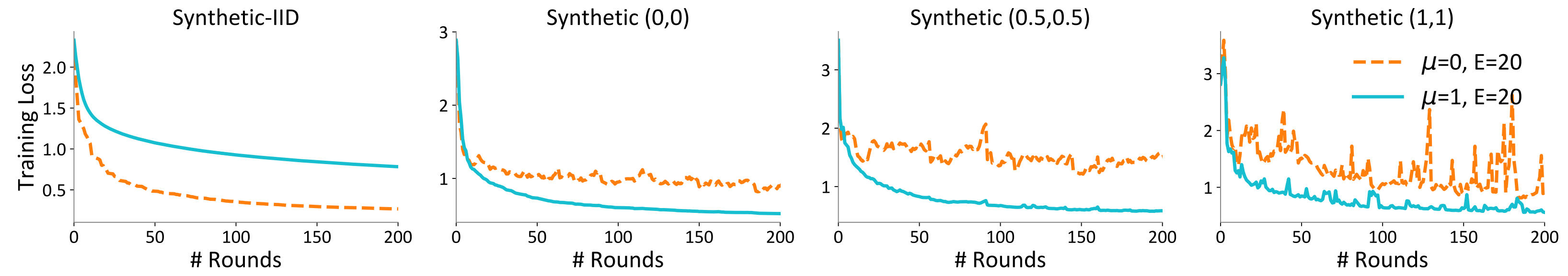
LEAF: A Benchmark for Learning in Federated Settings (website: leaf.cmu.edu)



Effects of Idea 1 (partial work):
Compare - - - - with ......
allowing for variable amounts of work to be performed can help convergence in the presence of systems heterogeneity

Effects of Idea 2 (the proximal term):
Compare ——— with ......
$\mu > 0$ leads to more stable convergence and enables otherwise divergent methods to converge

Increasing statistical heterogeneity leads to worse convergence; Setting $\mu > 0$ can help to combat this



Future Work

- How to tune μ automatically (hyper-parameter optimization for federated learning)?
- Can we quantify the statistical heterogeneity a priori and leverage it for improved performance?
- Better privacy metrics and mechanisms for federated learning?
- ……

Federated Learning: Challenges, Methods, and Future Directions (Signal Processing Magazine, arxiv.org/abs/1908.07873)

Code & Manuscript: www.cs.cmu.edu/~litian/; **Github:** github.com/litian96/FedProx