

# FEDERATED LEARNING

---

SUMMARY OF  
MODERN FEDERATED LEARNING METHODS

## REFERENCES

- ▶ Reddi, Sashank, et al. "Adaptive Federated Optimization." arXiv preprint arXiv:2003.00295 (2020).
- ▶ Wang, Hongyi, et al. "Federated learning with matched averaging." arXiv preprint arXiv:2002.06440 (2020).
- ▶ Xie, Cong, et al. "Generalized byzantine-tolerant sgd." arXiv preprint arXiv:1802.10116 (2018).
- ▶ Zhang, Jingzhao, et al. "Why ADAM beats SGD for attention models." arXiv preprint arXiv:1912.03194 (2019).
- ▶ Yurochkin, Mikhail, et al. "Bayesian nonparametric federated learning of neural networks." arXiv preprint arXiv:1905.12022 (2019).
- ▶ Wang, Yushi. "Co-op: Cooperative machine learning from mobile devices." (2017).
- ▶ Xie, Cong, Sanmi Koyejo, and Indranil Gupta. "Zeno++: Robust Fully Asynchronous SGD." arXiv preprint arXiv:1903.07020 (2019).
- ▶ Lalitha, Anusha, et al. "Peer-to-peer federated learning on graphs." arXiv preprint arXiv:1901.11173 (2019).

# CONTENTS

- ▶ 1. What is
  - ▶ Federated Learning
  - ▶ FedAvg
- ▶ 2. Problems on Federated Learning
- ▶ 3. Solutions
  - ▶ Adaptive Optimizer
  - ▶ Matched Averaging
  - ▶ Byzantine-tolerant
- ▶ 4. TODO

1-1.

# FEDERATED LEARNING

# FEDERATED LEARNING

- ▶ 연합 학습(Federated Learning, FL)
  - ▶ 분산 머신 러닝 패러다임
  - ▶ 많은 수의 클라이언트들이 중앙 서버에게
  - ▶ 자신의 훈련 데이터를 공유하지 않고 학습에 협력

# FEDERATED LEARNING

- ▶ **클라이언트**들이 모델을 업데이트
  - ▶ 임의의 클라이언트 최적화기를 이용해
  - ▶ 로컬 데이터의 손실(loss)을 줄이기 위해
- ▶ **서버**는 글로벌 모델을 업데이트
  - ▶ 클라이언트가 송신한 모델 업데이트의 평균으로부터
    - ▶ 로컬 모델의 파라미터들을 element-wise하게 평균
  - ▶ 클라이언트들에 걸친 손실을 최소화하기 위해

1-2.  
**FEDAVG**

## FEDAVG

- ▶ FEDAVG 기법
  - ▶ 각 라운드에서 클라이언트들의 서브셋이 선택되고 (주로 무작위로)
  - ▶ 서버가 그들에게 글로벌 모델을 브로드캐스트
  - ▶ 병렬적으로 클라이언트들이 고유의 손실 함수를 통해 SGD를 수행
  - ▶ 서버에게 **모델**을 전송
  - ▶ 서버는 로컬 모델들의 평균을 구해 글로벌 모델을 업데이트



## FEDAVG

- ▶ FEDAVG 기법의 재작성
  - ▶ 각 라운드에서 클라이언트들의 서브셋이 선택되고 (주로 무작위로)
  - ▶ 서버가 그들에게 글로벌 모델을 브로드캐스트
  - ▶ 병렬적으로 클라이언트들이 고유의 손실 함수를 통해 SGD를 수행
  - ▶ 서버에게 **델타(모델의 업데이트 정보)**를 전송
  - ▶ 서버는 델타들의 평균을 구해 글로벌 모델을 업데이트

# FEDAVG

- ▶ FEDAVG 기법의 재작성
- ▶ 수도-그라디언트 관점에서 일반화

---

## Algorithm 2 Generalized FEDAVG

---

Initialization:  $x_0$

**for**  $t = 0, \dots, T - 1$  **do**

    Sample subset  $\mathcal{S}$  of clients

$x_{i,0}^t = x_t$

**for** each client  $i \in \mathcal{S}$  **in parallel do**

**for**  $k = 0, \dots, K - 1$  **do**

            Compute an unbiased estimate  $g_{i,k}^t$  of  $\nabla F_i(x_{i,k}^t)$

$x_{i,k+1}^t = \text{CLIENTOPT}(x_{i,k}^t, g_{i,k}^t, \eta, t)$

$\Delta_i^t = x_{i,K}^t - x_t$

$\Delta_t = \frac{1}{|\mathcal{S}|} \sum_{i \in \mathcal{S}} \Delta_i^t$

$x_{t+1} = \text{SERVEROPT}(x_t, -\Delta_t, \eta, t)$

---

2.

**PROBLEMS**

## PROBLEMS ON FEDERATED LEARNING

- ▶ FL의 문제점
  - ▶ 이종적인 데이터
  - ▶ 높은 통신 비용
  - ▶ 클라이언트 드리프트
  - ▶ 적응형 학습률의 부재

## PROBLEMS ON FEDERATED LEARNING

- ▶ 이에 FL을 위한 많은 최적화 기법은
  - ▶ 로컬(local) 클라이언트 업데이트를 활용
  - ▶ 클라이언트가 통신 전에 로컬 모델을 수 차례 업데이트
- ▶ FL에 널리 쓰이는 로컬 최적화 기법 중 하나인
  - ▶ FEDAVG

## PROBLEMS ON FEDERATED LEARNING

- ▶ FEDAVG의 각 라운드에서
  - ▶ 클라이언트의 일부는 병렬적으로 SGD의 몇 에폭을 수행
  - ▶ 클라이언트 드리프트(client drift) 발생
- ▶ 클라이언트는 모델 업데이트 정보를 서버와 통신
  - ▶ 서버는 평균을 통해 새 글로벌 모델을 계산
  - ▶ 적응적 학습률(adaptive learning rate)의 부재

## CLIENT DRIFT

- ▶ 클라이언트 드리프트 (client drift)
  - ▶ 이종적 환경에서 여러 로컬 SGD 에폭의 수행은
  - ▶ 글로벌한 최적 모델로부터 멀리 떨어지게(drift) 만듦

## ADAPTIVE LEARNING RATE

- ▶ 적응적 학습률 (adaptive learning rate)
  - ▶ 학습 과정 중의 Heavy-tail 노이즈 분포 등
  - ▶ 필연적으로 적응적 학습률이 필요한 환경들이 있음
  - ▶ 이는 과거의 반복(iteration)들로부터 정보에 기반한
  - ▶ 그래디언트 기반의 최적화를 수행하도록 함
- ▶ 정보 공유를 하지 않는 FL의 기본 속성에 따르면
  - ▶ 적응형 학습률은 도전적인 문제임



3-1.

# ADAPTIVE OPTIMIZER

## ADAPTIVE OPTIMIZER

- ▶ 클라이언트와 서버의 학습률을 분리하는 자연스러운 방법을 제시
  - ▶ 즉, FEDAVG의 일반화
  - ▶ FEDAVG는 클라이언트와 서버가 SGD를 쓰고, 서버의 학습률이 1인 경우
- ▶ 클라이언트와 서버가 다른 학습률을 활용할 수 있도록 함
  - ▶ 로컬 업데이트의 억제 & 강화가 가능
  - ▶ 클라이언트 드리프트를 해결할 방법을 제공

## ADAPTIVE OPTIMIZER

- ▶ 적응형 방법들인
  - ▶ ADAGRAD, ADAM, YOGI 등은 머신 러닝 커뮤니티에서 유명한 기법들
  - ▶ 학습률을 튜닝할 수고를 덜고
  - ▶ Heavy-tail 노이즈 분포 등을 잘 다룸
- ▶ 이들 프레임워크 위에서
  - ▶ FL을 위한 적응형 최적화 기술들을 개발
  - ▶ 최초의 FL을 위한 적응형 최적화 기법

# EVALUATION

▶ 적응형 최적화기만 사용해도 FL의 성능이 크게 오름

FED...	ADAGRAD	ADAM	YOGI	AVGM	AVG
CIFAR-100	23.9	<b>42.3</b>	41.6	37.3	26.5
EMNIST CR	<b>85.7</b>	<b>86.0</b>	<b>86.1</b>	<b>86.3</b>	85.9
SHAKESPEARE	57.1	<b>57.4</b>	<b>57.6</b>	<b>57.5</b>	57.0
STACKOV... NWP	11.3	<b>22.1</b>	<b>22.2</b>	13.7	9.5
STACKOV... LR	<b>0.68</b>	0.62	0.64	0.22	0.19
EMNIST AE	7.29	16.99	<b>0.98</b>	<b>1.21</b>	2.63

3-2.

**MATCHED AVERAGING**

## MATCHED AVERAGING

- ▶ FedAvg에서 coordinate-wise averaging
  - ▶ 여러 문제가 생길 수 있음
  - ▶ 이는 신경망 파라미터의 **치환 불변성** 때문

## PROBABILITY FEDERATED NEURAL MATCHING

- ▶ 확률적(Probability) Federated Neural Matching (PFNM)
  - ▶ Averaging 하기 전
  - ▶ 클라이언트 신경망의 뉴런들을 매칭함으로써 문제를 다룸
  - ▶ 그러나 PFNM은 단순한 신경망에서만 동작
    - ▶ 가령, 전연결 feedforward 네트워크

## PROBABILITY FEDERATED NEURAL MATCHING

- ▶ PFNM을 CNN과 LSTM에 적용
  - ▶ 그러나 가중치 평균 방법 대비 매우 적은 수준의 향상만 있었음
- ▶ PFNM을 CNNs와 LSTMs로 확장할 수 있지만
  - ▶ 깊은 구조들에서 효과를 보지 못함
  - ▶ 반복적인 matched averaging이
  - ▶ 전체적으로 좋지 못한 결과를 야기한 것으로 추정



## MATCHED AVERAGING

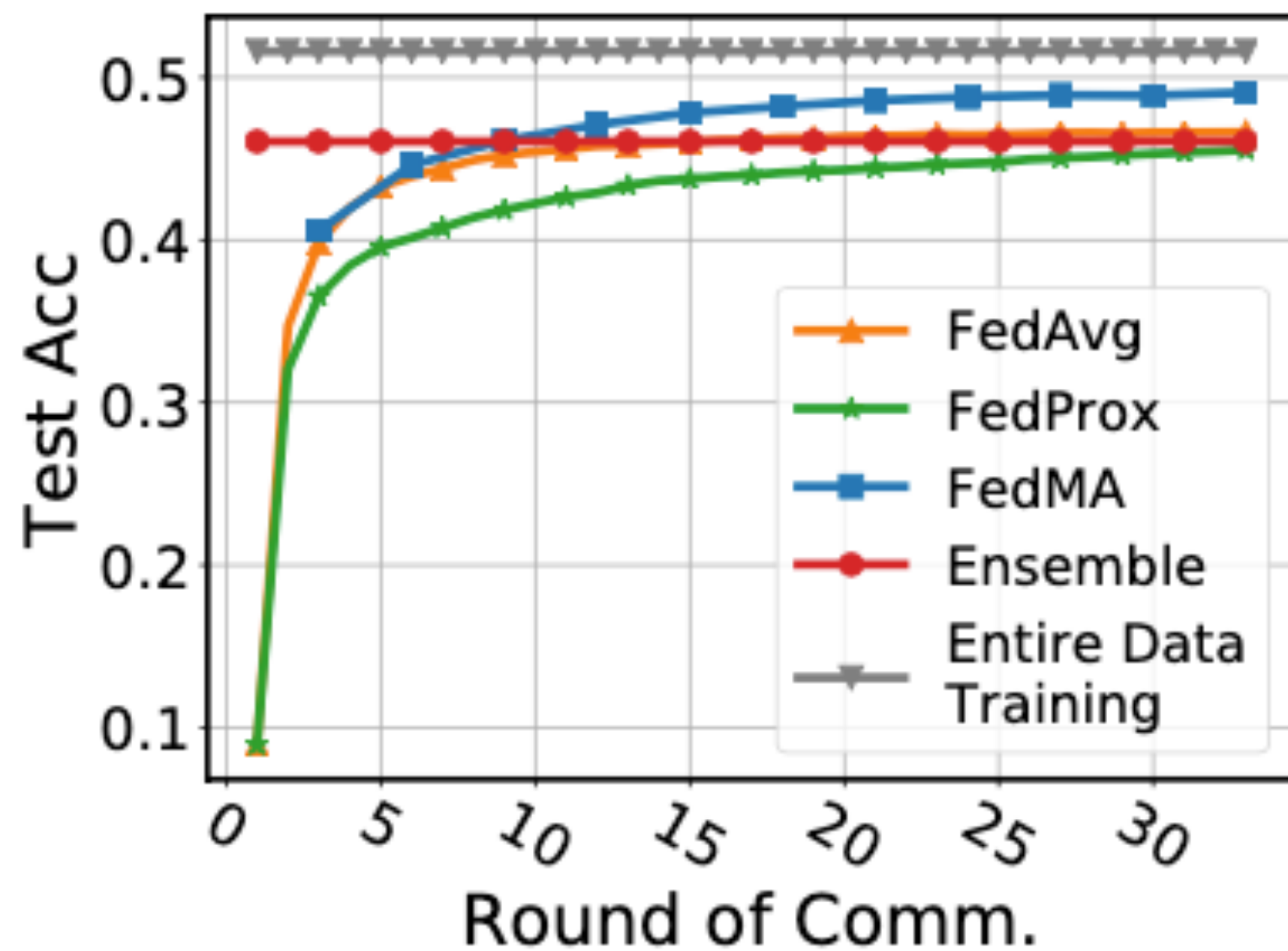
- ▶ Federated Matched Averaging (FedMA) 방법을 제안
  - ▶ 최신 CNN이나 LSTM을 위한 새로운 **layer-wise 연합 학습 알고리즘**
  - ▶ 커뮤니케이션 비용을 줄이고, SOTA의 성능을 보임

## MATCHED AVERAGING

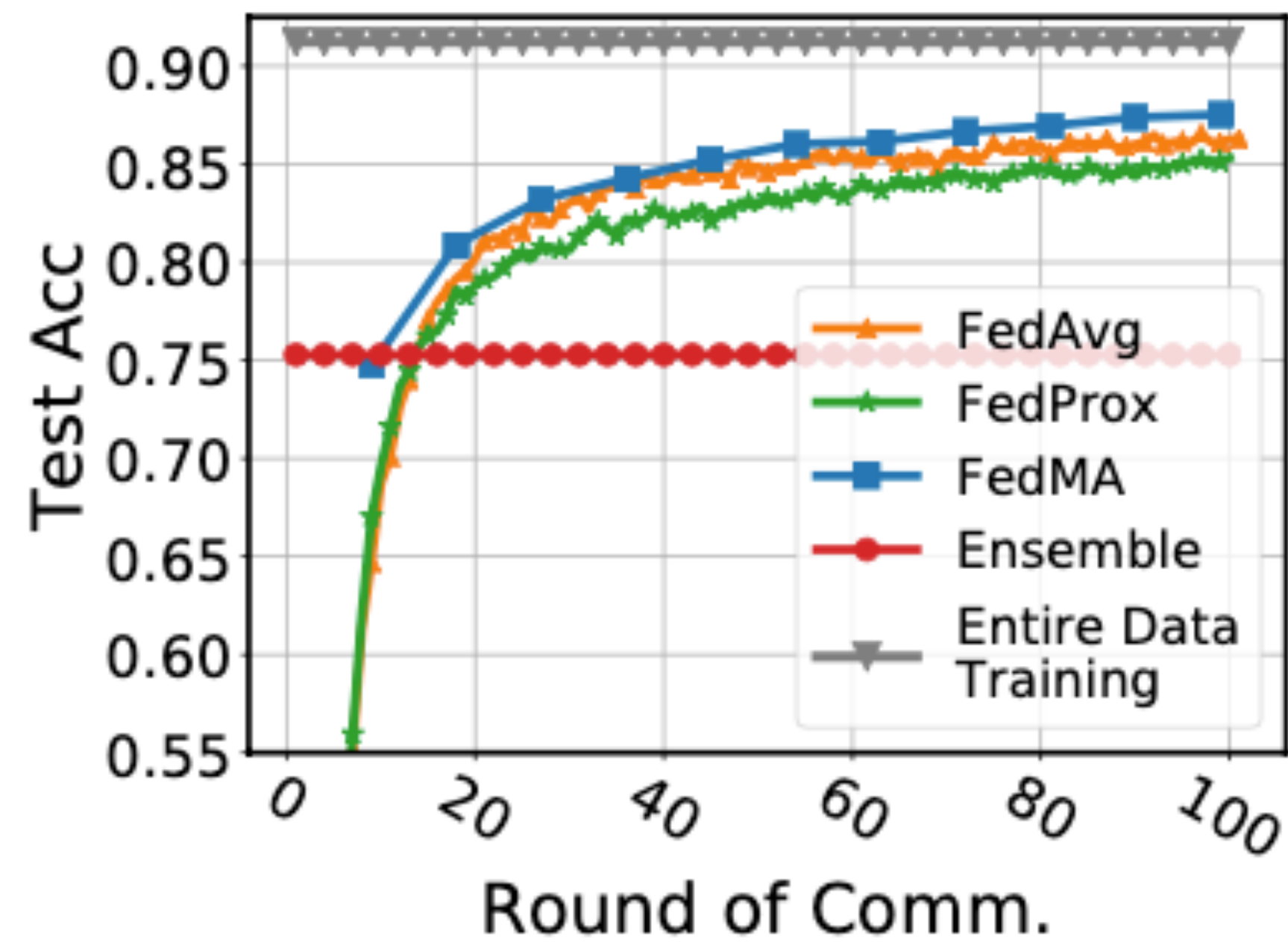
- ▶ Layer-wise한 매칭 스킴을 제안
  - ▶ 데이터 센터(서버)가 클라이언트로부터 **레이어의 첫 층**에 해당하는 가중치를 수집
  - ▶ 한 레이어에 해당하는 매칭을 수행
  - ▶ 연합 모델의 가중치를 클라이언트들에게 브로드캐스트
  - ▶ 클라이언트는 반영 후 연속되는 모든 레이어를 고유 데이터셋으로 학습
    - ▶ 이 때 연합 레이어는 동결함
  - ▶ **마지막 레이어**에 이르기까지 반복

## EVALUATION

- ▶ FedMA가 FedAvg와 FedProx 성능을 상회



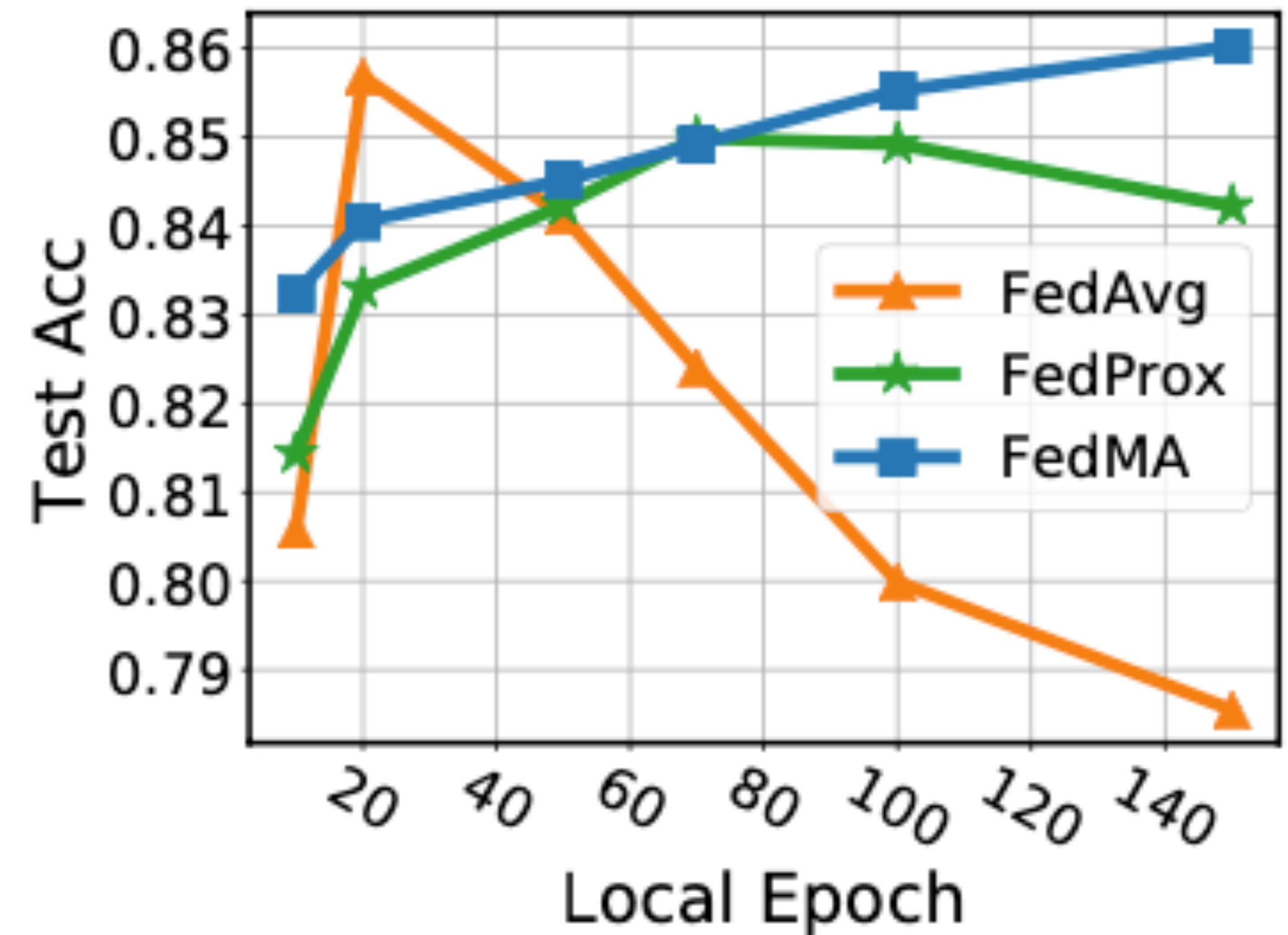
(b) LSTM, Shakespeare; rounds



(d) VGG-9, CIFAR-10; rounds

## EVALUATION

- ▶ 사전 연구들에 따르면
- ▶ 로컬(local) 학습에서의 에폭  $E$ 에 따라
  - ▶ FedAvg의 성능에 영향을 끼침은 물론
  - ▶ 때로는 발산하는 결과를 야기
- ▶ 그러나 FedMA에서는
  - ▶ Local Epoch이 클 수록 유리
  - ▶ 로컬 클라이언트가 원하는 만큼 학습할 수 있는 **오직 유일한 방법**



3-3.

**BYZANTINE TOLERANT**

## GENERALIZED BYZANTINE

- ▶ 분산 학습에서, 여러 종류의 공격 유형이 있음
- ▶ 일반적으로 공격자는 모델 학습을 방해하고자 함
  - ▶ SGD 수렴을 느리게 만들거나
  - ▶ 나쁜 솔루션을 향하게 함
- ▶ 각 차원에 대해 비잔틴 값의 수는 절반 미만으로 가정
  - ▶ 흔한 가정
  - ▶ Dimensional Byzantine 탄력성이라 칭함

## MEDIAN-BASED AGGREGATION

- ▶ 3개의 중앙값 기반 통합 규칙을 제안
  - ▶ Geometric Median
  - ▶ Marginal Median
  - ▶ Beyond Median

## MEDIAN-BASED AGGREGATION

- ▶ Geometric Median (기하중앙값)
- ▶ 평균에 대한 **강건한** 추정량으로 사용
  - ▶ 최대, 데이터의 절반이 부정해도
  - ▶ 부정하지 않은 데이터에 대한 추정을 제공



## MEDIAN-BASED AGGREGATION

- ▶ Marginal Median
- ▶  $\mu = \text{MarMed}(\{\tilde{v}_i : i \in [n]\})$
- ▶  $\mu$ 의  $j$ 번째 차원은  $\mu_j = \text{median}(\{(\tilde{v}_1)_j, \dots, (\tilde{v}_n)_j\})$ 
  - ▶  $\text{median}(\cdot)$ 은 1차원 중앙값

## MEDIAN-BASED AGGREGATION

- ▶ Beyond Median
- ▶ 비잔틴의 수  $q$ 를 쉽게 추정할 수 있다면
- ▶ 중앙값에 가까운  $n - q$  개의 값의 평균을 활용할 수 있을 것
  - ▶ “mean around median”

## EVALUATION

- ▶ 예상대로, 평균(mean) 방법은 비잔틴 탄력성이 없음
- ▶ GeoMed
  - ▶ 전통적인 비잔틴 탄력성은 있으나
  - ▶ Dimensional 비잔틴 탄력성은 없음
- ▶ MarMed와 MeaMed는 Dimensional 비잔틴 탄력성이 있음
  - ▶ 그러나 Omniscient 공격(SGD의 방향을 크게 반대로)에서 MarMed는 수렴이 늦음

4.

TODO

# TODO

- ▶ 네트워크의 일반화
  - ▶ 클라이언트-서버 구조가 아닌
  - ▶ P2P 등 Decentralized & Asynchronous 한 환경
  - ▶ (+) Byzantine
- ▶ Co-op
  - ▶ Cooperative Learning
- ▶ Zeno++