Federated Meta-Learning with Fast Convergence and Efficient Communication

Fei Chen*
Huawei Noah's Ark Lab
chen.f@huawei.com

Mi Luo*
Huawei Noah's Ark Lab
rosemaryluo@outlook.com

Zhenhua Dong
Huawei Noah's Ark Lab
dongzhenhua@huawei.com

Zhenguo Li
Huawei Noah's Ark Lab
li.zhenguo@huawei.com

Xiuqiang He
Huawei Noah's Ark Lab
hexiuqiang1@huawei.com

arXiv(2018)
Cite: 131

2021.06.23
임진혁

Index

임진혁<eeplearning@units.ac.kr>

Introduction

This paper show
“meta-learning is a natural choice to handle federate issues”

In short ,

Proposed method (FedMETA) is MAML of the federated setting

Limitation of Distributed Learning Setting

- **Statistical Challenges**

  - decentralized data in non IID (highly personalized & heterogeneous)
  - results significant reduction in model accuracy

- **Systematic Challenges**

  - number of devices is larger than traditional distributed settings
  - each device constraint: storage / computation / communication

임진혁<eeplearning@units.ac.kr>

Introduction

FedAVG

- **flexibly determine the number of epochs and batch size** for local training(SGD)
- can achieve high model accuracy as well as trade-off between **computation & communication**

MAML

- rapid adaption and good generalization to new task
(slowly learned from a large number of tasks through meta-training process)
(to "model" is fast trained for each new task)
- Task : consist of
support Set : model trained
quary Set : tested
the tested results are used to update "model"

임진혁<eeplearning@units.ac.kr>

2 Proposed Method

FedMETA

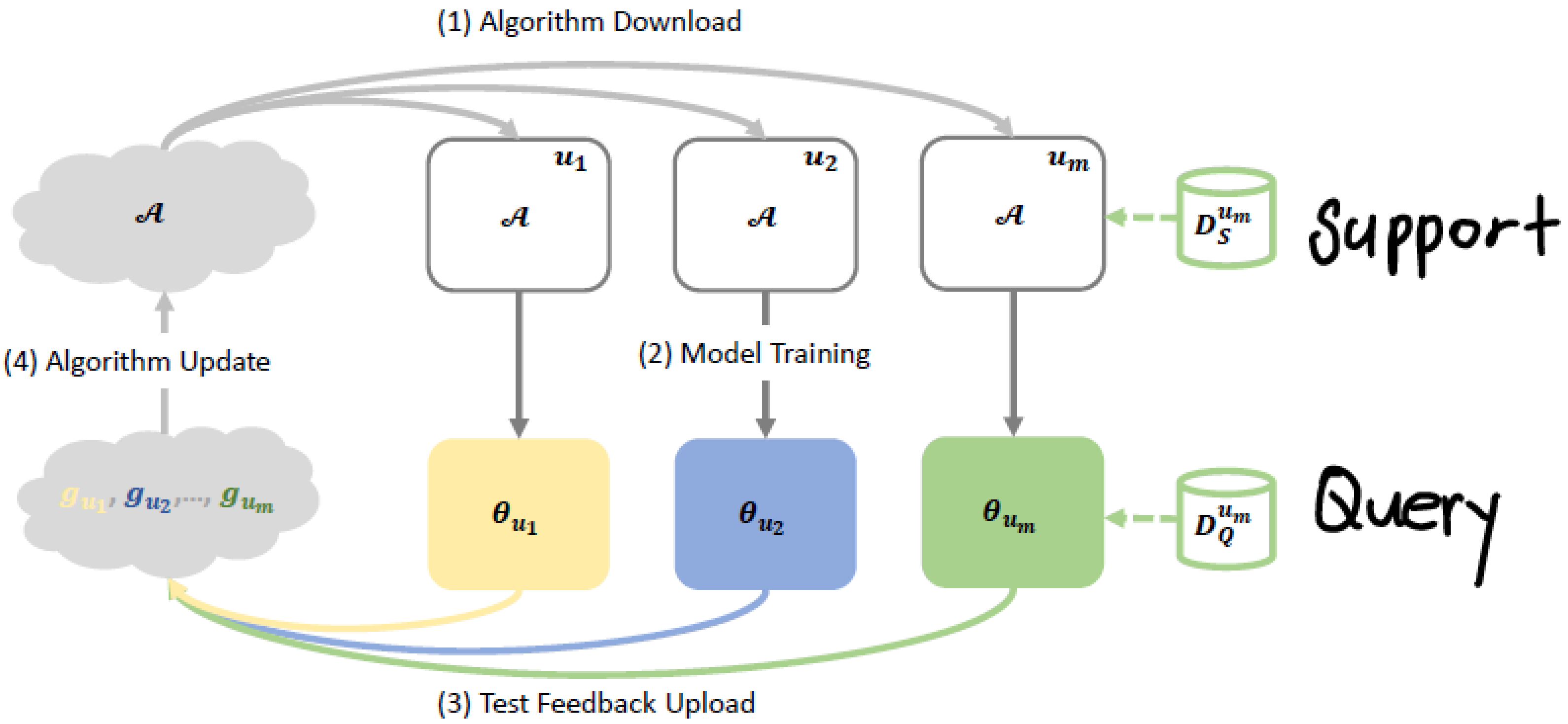


Figure 1: Workflow of the federated meta-learning framework.

2 Proposed Method

FedMETA

Algorithm 1: `FedMeta` with MAML and Meta-SGD

// Run on the server

AlgorithmUpdate:

Initialize θ for MAML, or initialize (θ, α) for Meta-SGD.

for each episode $t = 1, 2, \ldots$ **do**

  Sample a set U_t of m clients, and distribute θ (for MAML) or (θ, α) (for Meta-SGD) to the sampled clients.

  for each client $u \in U_t$ in parallel **do**

    Get test loss $g_u \leftarrow \text{ModelTrainingMAML}(\theta)$ or

    $g_u \leftarrow \text{ModelTrainingMetaSGD}(\theta, \alpha)$

  end

  Update algorithm paramters $\theta \leftarrow \theta - \frac{\beta}{m} \sum_{u \in U_t} g_u$ for MAML or

  $(\theta, \alpha) \leftarrow (\theta, \alpha) - \frac{\beta}{m} \sum_{u \in U_t} g_u$ for Meta-SGD.

end

// Run on client u

ModelTrainingMAML(θ)**:**

Sample support set D_S^u and query set D_Q^u

$\mathcal{L}_{D_S^u}(\theta) \leftarrow \frac{1}{|D_S^u|} \sum_{(x,y) \in D_S^u} \ell(f_\theta(x), y)$

$\theta_u \leftarrow \theta - \alpha \nabla \mathcal{L}_{D_S^u}(\theta)$

$\mathcal{L}_{D_Q^u}(\theta_u) \leftarrow \frac{1}{|D_Q^u|} \sum_{(x',y') \in D_Q^u} \ell(f_{\theta_u}(x'), y')$

$g_u \leftarrow \nabla_\theta \mathcal{L}_{D_Q^u}(\theta_u)$

Return g_u to server

ModelTrainingMetaSGD(θ, α)**:**

Sample support set D_S^u and query set D_Q^u

$\mathcal{L}_{D_S^u}(\theta) \leftarrow \frac{1}{|D_S^u|} \sum_{(x,y) \in D_S^u} \ell(f_\theta(x), y)$

$\theta_u \leftarrow \theta - \alpha \circ \nabla \mathcal{L}_{D_S^u}(\theta)$

$\mathcal{L}_{D_Q^u}(\theta_u) \leftarrow \frac{1}{|D_Q^u|} \sum_{(x',y') \in D_Q^u} \ell(f_{\theta_u}(x'), y')$

$g_u \leftarrow \nabla_{(\theta,\alpha)} \mathcal{L}_{D_Q^u}(\theta_u)$

Return g_u to server

Experiments

- **FedMeta**

- **FedAVG(meta)**

- **MetaSGD**

FedMeta provide faster convergence, higher accuracy and lower system overhead

임진혁<eeplearning@units.ac.kr>

Experiments on LEAF Dataset (benchmark for federated setting)

Table 1: Statistics of selected datasets.

Dataset	Clients	Samples	Classes	samples per client		classes per client	
				mean	stdev	min	max
FEMNIST	1,068	235,683	62	220	90	9	62
Shakespeare	528	625,127	70	1183	1218	2	70
Sent140	3,790	171,809	2	45	28	1	2

(1) FEMNIST : CNN

- 62 Class Classification (Complex version of MNIST)
- Partitioned on the writer of the Digit / Character

(2) **Shakespeare**: Char-LSTM

- Partitioned on each speaking role

(3) **Sentiment140**: LSTM-Classifier

- Binary sentiment classification (generated by annotating tweets)
- Partitioned on each twitter user

임진혁<eeplearning@units.ac.kr>

Experiments on LEAF Dataset (benchmark for federated setting)

Table 1: Statistics of selected datasets.

Dataset	Clients	Samples	Classes	samples per client		classes per client	
				mean	stdev	min	max
FEMNIST	1,068	235,683	62	220	90	9	62
Shakespeare	528	625,127	70	1183	1218	2	70
Sent140	3,790	171,809	2	45	28	1	2

- Randomly select 80% of client as training clients
- 10% as validation clients, remaining 10% as testing clients
- Each client, data is divided into “support set” and “query set”
- Fraction p: “support set” using percentage for each client
  (To evaluate how Algorithms adapt to new users with limited data”)
- Filter inactive clients with fewer than k records
  ([1]:10 ,[2]:20, [3]:23)
- For considering the limited computation capacity on edge device, local epochs => 1

임진혁<eeplearning@units.ac.kr>

3 Experiments

- FedMETA(MAML, MetaSGD) achieve increase in the **final accuracy** with **faster** and **more stable convergence**

  (similar performance but in case of Sent140, SGD better)

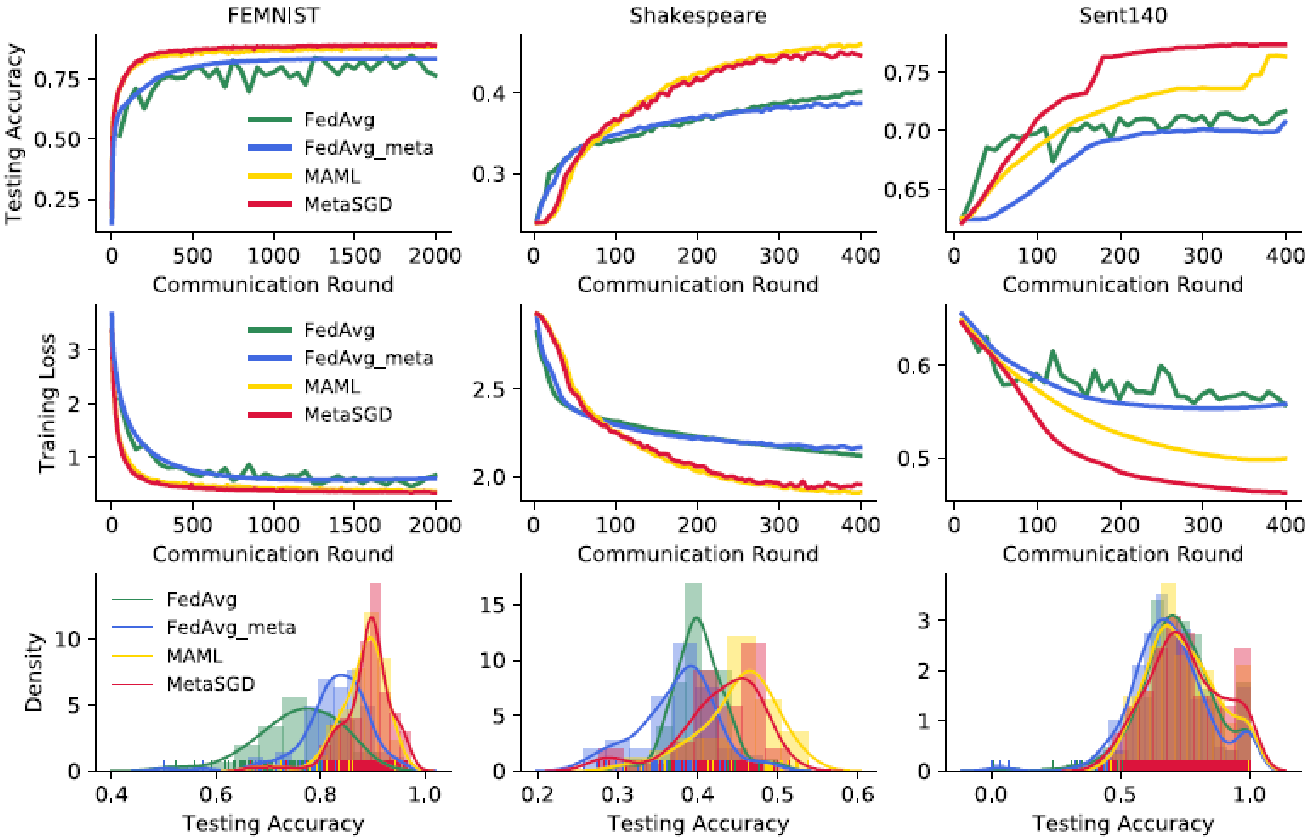


Figure 2: Performance on LEAF datasets for FedAvg and three running examples of `FedMeta`. The support fraction setting for all datasets is 20%. Compared with intuitive FedAvg, all the running examples within `FedMeta` framework provide faster convergence and higher accuracy.

Final accuracies after several rounds

FEMNIST: 2000 round
Shakespeare: 400 round
Sent140: 400 round

Increasing accuracy by 3.23% - 14.84%

In the CASE of 20(p) support(NLP):
model suffers excessive deviation from the global optimal
afrer finetuning on a small amount data

Table 2: Accuracy results on LEAF Datasets. For FEMNIST, Shakespeare and Sent140, the models are trained for 2000, 400 and 400 rounds respectively.

		20% Support	50% Support	90% Support
FEMNIST	FedAvg	76.79% ± 0.45%	75.44% ± 0.73%	77.05% ± 1.43%
	FedAvg(Meta)	83.58% ± 0.13%	87.84% ± 0.11%	88.76% ± 0.78%
	FedMeta(MAML)	88.46% ± 0.25%	89.77% ± 0.08%	89.31% ± 0.15%
	FedMeta(Meta-SGD)	**89.26%** ± 0.12%	**90.28%** ± 0.02%	**89.31%** ± 0.09%
Shakespeare	FedAvg	40.76% ± 0.62%	42.01% ± 0.43%	40.58% ± 0.55%
	FedAvg(Meta)	38.71% ± 0.51%	42.97% ± 0.97%	43.48% ± 0.64%
	FedMeta(MAML)	**46.06%** ± 0.85%	**46.29%** ± 0.84%	**46.49%** ± 0.77%
	FedMeta(Meta-SGD)	44.72% ± 0.72%	45.24% ± 0.53%	46.25% ± 0.63%
Sent140	FedAvg	71.53% ± 0.18%	72.29% ± 0.49%	73.38% ± 0.38%
	FedAvg(Meta)	70.10% ± 0.66%	73.88% ± 0.06%	75.86% ± 0.46%
	FedMeta(MAML)	76.37% ± 0.06%	78.63% ± 0.19%	79.53% ± 0.25%
	FedMeta(Meta-SGD)	**77.24%** ± 0.32%	**79.38%** ± 0.09%	**80.94%** ± 0.29%

임진혁<eeplearning@units.ac.kr>

Final accuracies after several rounds

FEMNIST: 2000 round
Shakespeare: 400 round
Sent140: 400 round

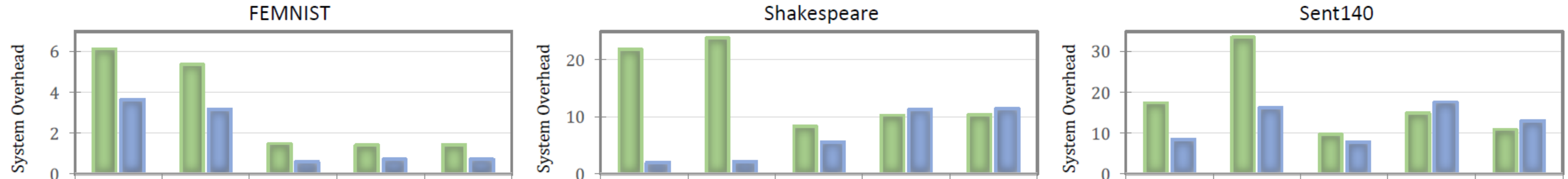
WHEN Increasing p(support):
(20% to 90%)
FedMETA's increasing is very small

Better Generalization ability to new clients with limited data
(not only lead to higher mean accuracies,
but also achieve more centered accuracy distribution with lower variance)

Table 2: Accuracy results on LEAF Datasets. For FEMNIST, Shakespeare and Sent140, the models are trained for 2000, 400 and 400 rounds respectively.

		20% Support	50% Support	90% Support
FEMNIST	FedAvg	76.79% ± 0.45%	75.44% ± 0.73%	77.05% ± 1.43%
	FedAvg(Meta)	83.58% ± 0.13%	87.84% ± 0.11%	88.76% ± 0.78%
	FedMeta(MAML)	88.46% ± 0.25%	89.77% ± 0.08%	89.31% ± 0.15%
	FedMeta(Meta-SGD)	**89.26%** ± 0.12%	**90.28%** ± 0.02%	**89.31%** ± 0.09%
Shakespeare	FedAvg	40.76% ± 0.62%	42.01% ± 0.43%	40.58% ± 0.55%
	FedAvg(Meta)	38.71% ± 0.51%	42.97% ± 0.97%	43.48% ± 0.64%
	FedMeta(MAML)	**46.06%** ± 0.85%	**46.29%** ± 0.84%	**46.49%** ± 0.77%
	FedMeta(Meta-SGD)	44.72% ± 0.72%	45.24% ± 0.53%	46.25% ± 0.63%
Sent140	FedAvg	71.53% ± 0.18%	72.29% ± 0.49%	73.38% ± 0.38%
	FedAvg(Meta)	70.10% ± 0.66%	73.88% ± 0.06%	75.86% ± 0.46%
	FedMeta(MAML)	76.37% ± 0.06%	78.63% ± 0.19%	79.53% ± 0.25%
	FedMeta(Meta-SGD)	**77.24%** ± 0.32%	**79.38%** ± 0.09%	**80.94%** ± 0.29%

System Overhead



FEMNIST

	FedAvg	FedAvg (Meta)	FOMAML	MAML	MetaSGD
Bytes (1e10)	6.15	5.37	1.5	1.42	1.44
Flops (1e12)	3.65	3.16	0.59	0.72	0.73

Shakespeare

	FedAvg	FedAvg (Meta)	FOMAML	MAML	MetaSGD
Bytes (1e9)	21.9	23.83	8.44	10.28	10.45
Flops (1e13)	2.03	2.20	5.61	11.31	11.48

Sent140

	FedAvg	FedAvg (Meta)	FOMAML	MAML	MetaSGD
Bytes (1e9)	17.49	33.57	9.74	14.91	10.91
Flops (1e11)	8.52	16.39	7.85	17.70	13.10

Figure 3: System overhead for achieving a target accuracy in different methods.The target accuracies for FEMNIST, Shakespeare and Sent140 are 74%, 38% and 70% respectively.

Total number of FLOPS[across all device]
Total number of bytes[uploaded to and downloaded from server all device]

⇒ FedMETA achieves reduction in “communication cost”

감사합니다

임진혁<eeplearning@unist.ac.kr>