Heterofl: Computation and communication efficient federated learning for heterogeneous clients

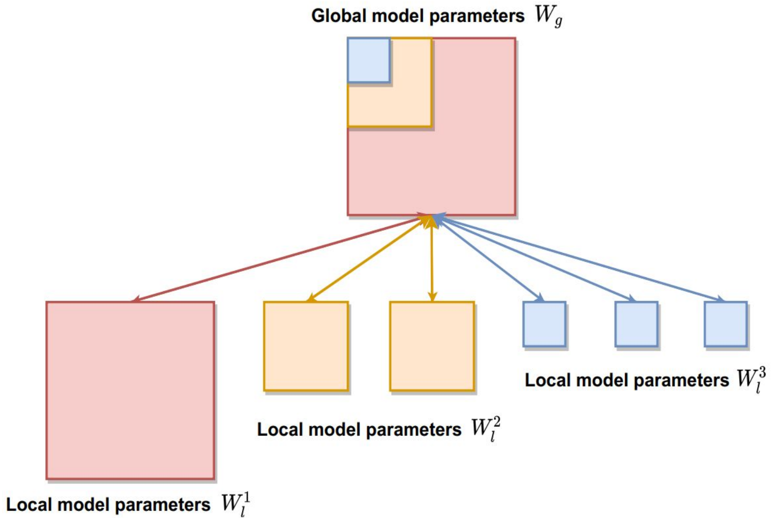
Enmao Diao, Jie Ding, Vahid Tarokh
Duke University, University of Minnesota-Twin Cities
ICLR 2020

概述

- 背景
  - 用户的计算和通信能力有所不同，甚至动态变化
- 贡献
  - 提出HeteroFL来训练各种各样的（heterogeneous）本地模型，得到一个全局模型

假设(Heterogeneous Models)

- 本地模型可以有相似的网络结构，但可以缩减复杂度
- 隐藏通道数不同
- 计算复杂度等级：$W_l^p \subset W_l^{p-1} \cdots \subset W_l^1$
- 处于中间复杂度的模型的参数被大模型完全覆盖，部分被小模型影响



$$W_g = W_l^1 = W_l^p \cup (W_l^{p-1} \setminus W_l^p) \cup \cdots \cup (W_l^1 \setminus W_l^2)$$

static BN & Scaler

- sBN
  - 不追踪运行数据，仅对批数据进行归一化
  - 服务器按序询问本地服务器，同时更新全局BN数据
- Scaler（不同计算复杂度的模型参数会偏离到不同规模）
  - 在训练阶段，缩放表达 $\frac{1}{r^{p-1}}$
  - 推理阶段，全局模型不进行缩放
- 整个流程

$$y = \phi(\mathrm{sBN}(\mathrm{Scaler}(X_m W_m^p + b_m^p)))$$

伪代码

Algorithm 1: HeteroFL: Heterogeneous Federated Learning

Input: Data X_i distributed on M local clients, the fraction C of active clients per communication round, the number of local epochs E, the local minibatch size B, the learning rate η, the global model parameterized by W_g, the channel shrinkage ratio r, and the number of computation complexity levels P.

System executes:

Initialize W_g^0 and local capabilities information $L_{1:K}$

for each communication round $t = 0, 1, 2, \ldots$ **do**

$M_t \leftarrow \max(C \cdot M, 1)$

$S_t \leftarrow$ random set of M_t clients

for each client $m \in S_t$ **in parallel do**

Determine computation complexity level p based on L_m

$r_m \leftarrow r^{(p-1)}$, $d_m \leftarrow r_m d_g$, $k_m \leftarrow r_m k_g$

$W_m^t \leftarrow W_g^t[: d_m, : k_m]$

$W_m^{t+1} \leftarrow \text{ClientUpdate}(m, r_m, W_m^t)$

end

for each computation complexity level p **do**

$W_g^{p-1,t+1} \setminus W_g^{p,t+1} \leftarrow \frac{1}{M_t - M_{p:P,t}} \sum_{i=1}^{M_t - M_{p:P,t}} W_i^{p-1,t+1} \setminus W_i^{p,t+1}$

end

$W_g^{t+1} \leftarrow \bigcup_{p=1}^{P} W_g^{p-1,t+1} \setminus W_g^{p,t+1}$

Update $L_{1:K}, \eta$ (Optional)

end

Query representation statistics from local clients (Optional)

ClientUpdate (m, r_m, W_m)**:**

$B_m \leftarrow$ Split local data X_m into batches of size B

for each local epoch e from 1 to E **do**

for batch $b_m \in B_m$ **do**

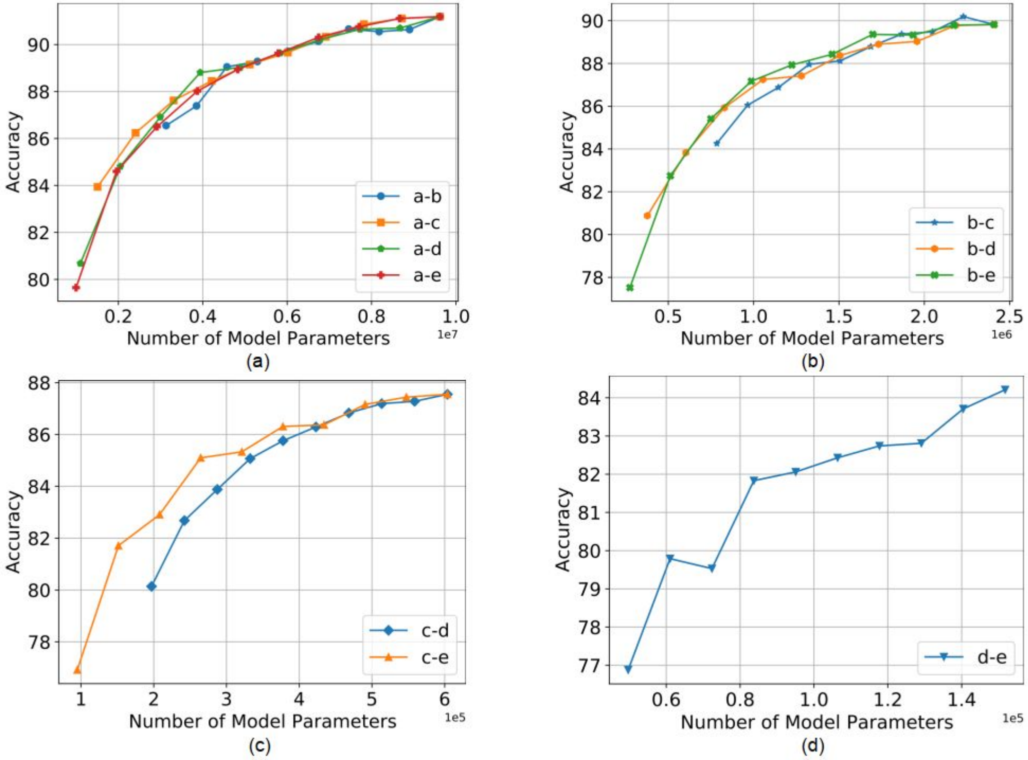
$W_m \leftarrow W_m - \eta \nabla \ell(W_m, r_m; b_m)$

end

end

Return W_m to server

实验结果



Accuracy
Number of Model Parameters
1e7
a-b
a-c
a-d
a-e
(a)
Accuracy
Number of Model Parameters
1e6
b-c
b-d
b-e
(b)
Accuracy
Number of Model Parameters
1e5
c-d
c-e
(c)
Accuracy
Number of Model Parameters
1e5
d-e
(d)

实验结果

Model	Ratio	Parameters	FLOPs	Space (MB)	Accuracy		
					IID	Non-IID	
						Local	Global
a	1.00	1.6 M	80.5 M	5.94	99.53	99.85	98.92
a-e	0.50	782 K	40.5 M	2.98	99.46	99.89	98.96
a-b-c-d-e	0.27	416 K	21.6 M	1.59	99.46	99.85	98.29
b	1.00	391 K	20.5 M	1.49	99.53	99.87	99.10
b-e	0.51	199 K	10.4 M	0.76	99.51	99.67	98.51
b-c-d-e	0.33	131 K	6.9 M	0.50	99.52	99.88	98.99
c	1.00	99 K	5.3 M	0.38	99.35	99.56	96.34
c-e	0.53	53 K	2.9 M	0.20	99.39	99.79	97.27
c-d-e	0.44	44 K	2.4 M	0.17	99.31	99.76	97.85
d	1.00	25 K	1.4 M	0.10	99.17	99.86	97.86
d-e	0.63	16 K	909 K	0.06	99.19	99.63	97.70
e	1.00	7 K	400 K	0.03	98.66	99.07	92.84
Standalone (Liang et al., 2020)	1.00	633 K	1.3 M	2.42	86.24	98.72	30.41
FedAvg (Liang et al., 2020)	1.00	633 K	1.3 M	2.42	97.93	98.20	98.20
LG-FedAvg (Liang et al., 2020)	1.00	633 K	1.3 M	2.42	97.93	98.54	98.17

总结

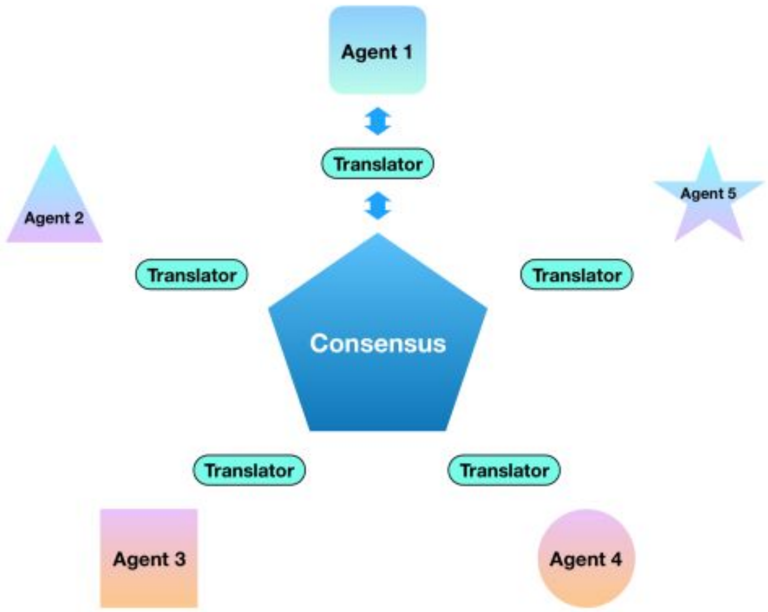
- 结果表明在iid和non.iid的情况下效果不错
- 规定了模型仅宽度不同

FedMD: Heterogenous Federated Learning via Model Distillation

Daliang Li, Junpu Wang
Harvard University, Yale University
NeurIPS workshop 2019

概述

- 背景
  - 因为隐私和知识产权，参与者不愿分享模型的 细节
- 成果
  - 提出FedMD，服务器无需控制模型架构
  - 在参与者间翻译知识

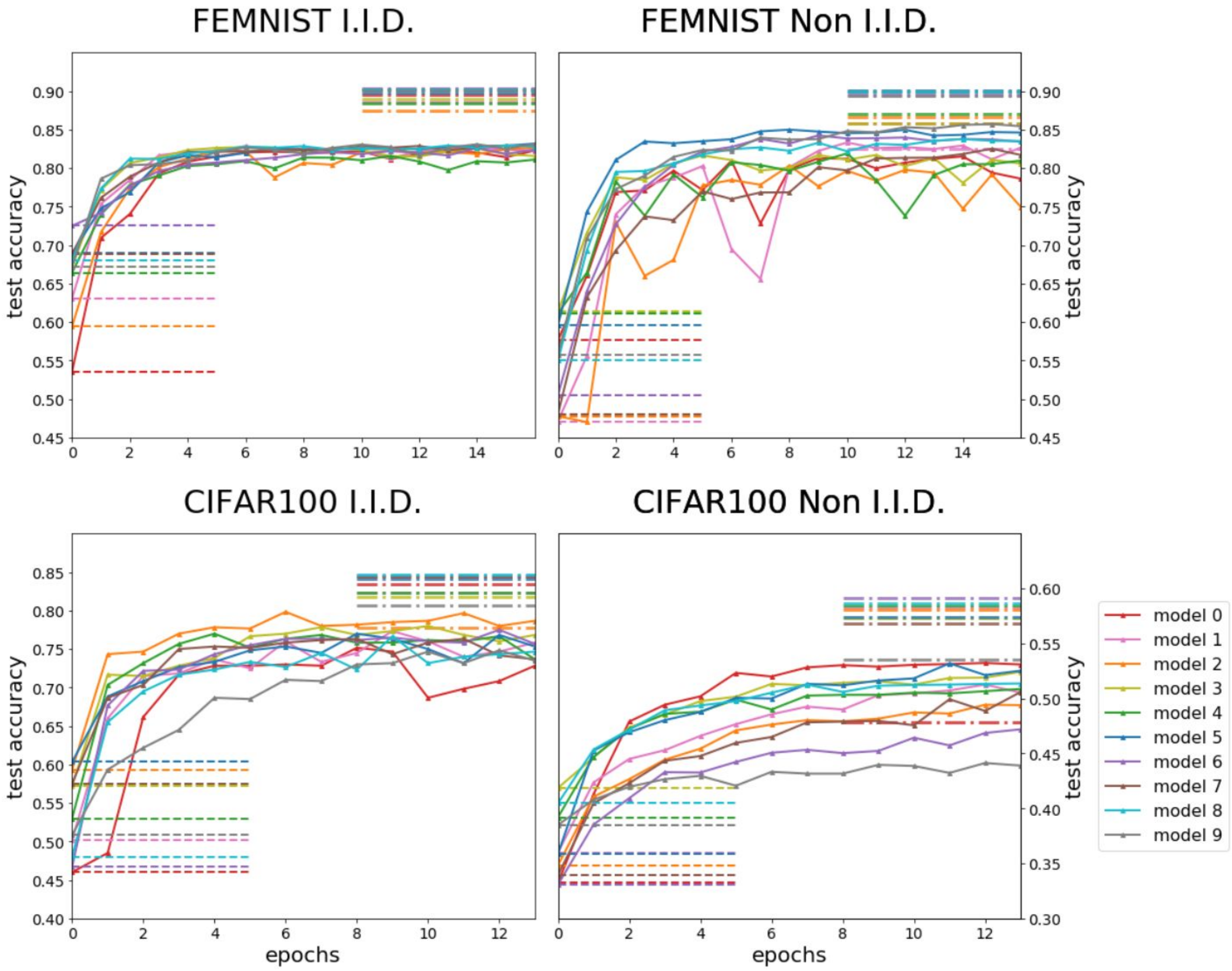


框架

Algorithm 1: The FedMD framework enabling federated learning for heterogeneous models.

Input: Public dataset $\mathcal{D}_0$, private datasets $\mathcal{D}_k$, independently designed model f_k, $k = 1 \dots m$,
Output: Trained model f_k
Transfer learning: Each party trains f_k to convergence on the public $\mathcal{D}_0$ and then on its private $\mathcal{D}_k$.
for j=1,2...P **do**
  Communicate: Each party computes the class scores $f_k(x_i^0)$ on the public dataset, and transmits the result to a central server.
  Aggregate: The server computes an updated consensus, which is an average $\tilde{f}(x_i^0) = \frac{1}{m}\sum_k f_k(x_i^0)$.
  Distribute: Each party downloads the updated consensus $\tilde{f}(x_i^0)$.
  Digest: Each party trains its model f_k to approach the consensus $\tilde{f}$ on the public dataset $\mathcal{D}_0$.
  Revisit: Each party trains its model f_k on its own private data for a few epochs.
end

实验结果



FEMNIST I.I.D.
FEMNIST Non I.I.D.
CIFAR100 I.I.D.
CIFAR100 Non I.I.D.
test accuracy
epochs
model 0
model 1
model 2
model 3
model 4
model 5
model 6
model 7
model 8
model 9

总结

- 工作较为简单
- 每个参与方要达到共识可能比较麻烦