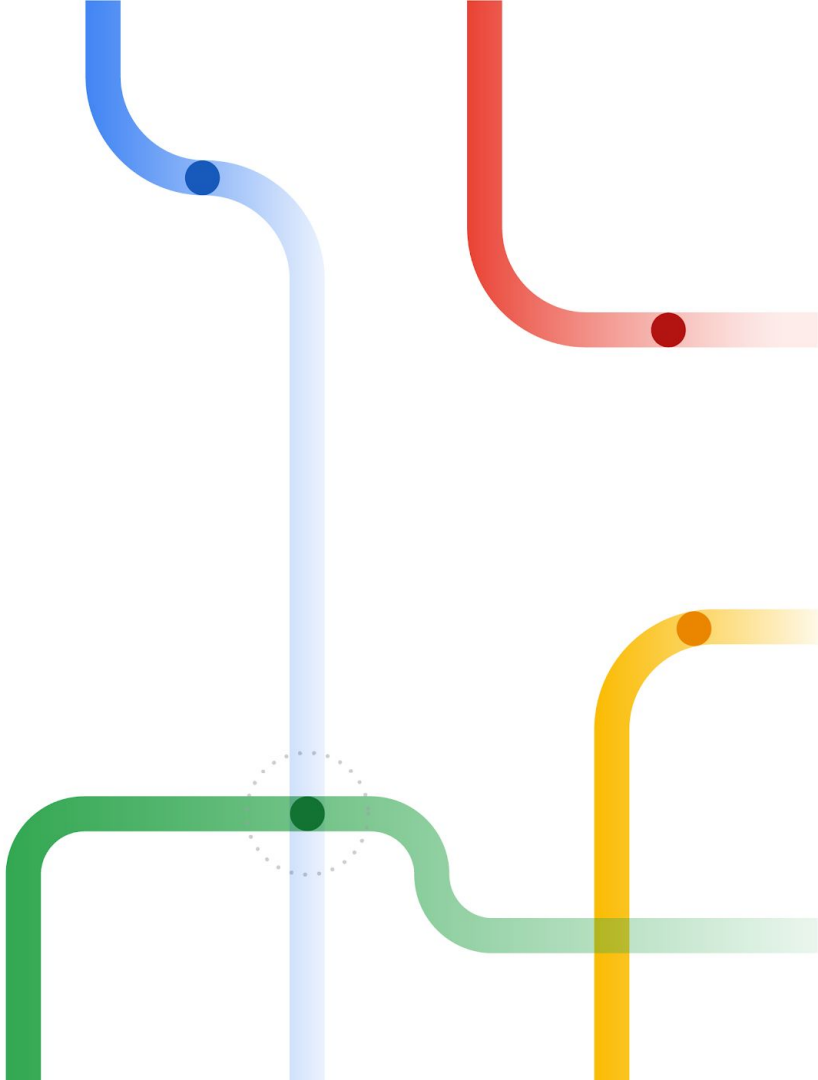
The Fundamental Price of Secure Aggregation in Differentially Private Federated Learning

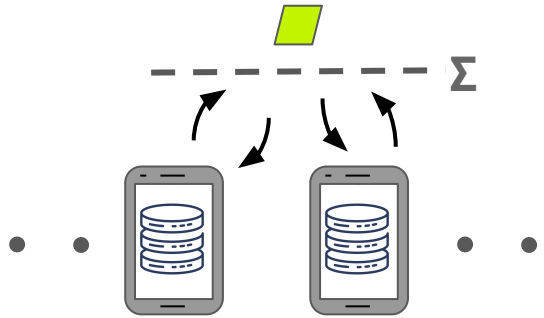
Christopher A. Choquette-Choo

With: Wei-ning Chen, Peter Kairouz, and Ananda Theertha Suresh

Google Research



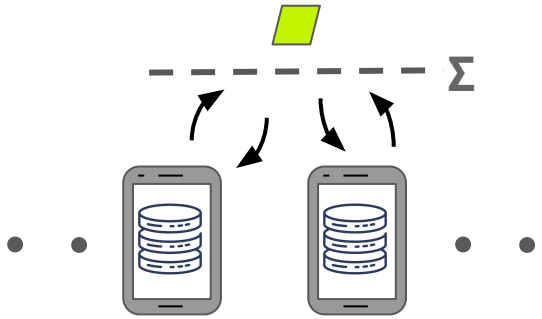
Federated Learning (FL)

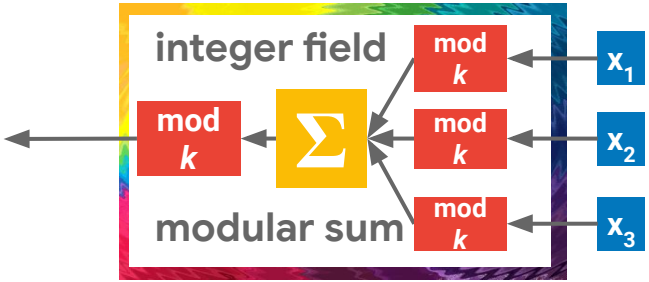


Train high utility models

Background

Federated Learning (FL) + **Secure Aggregation (SecAgg)**



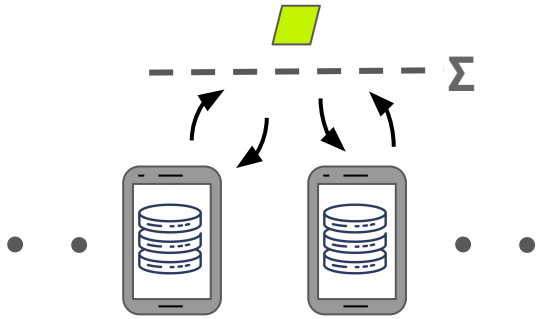


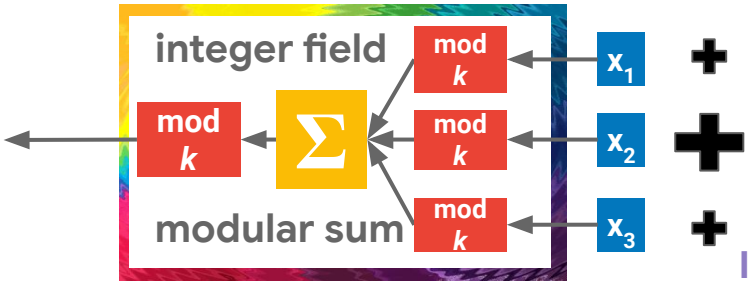
Train high utility models

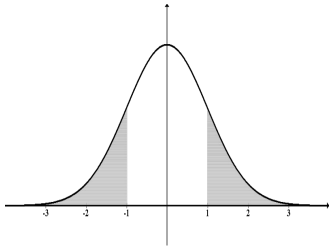
Security at the cost of increased communication

Background

Federated Learning (FL) + **Secure Aggregation (SecAgg)** + **Differential Privacy (DP)**







Injected distributively, i.e., distributed DP

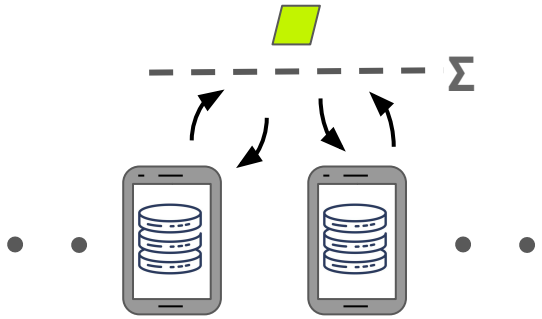
Train high utility models

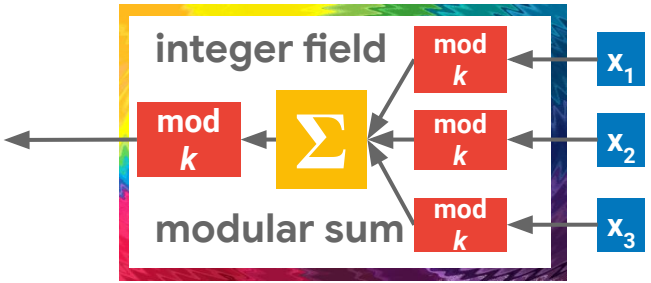
Security at the cost of increased communication

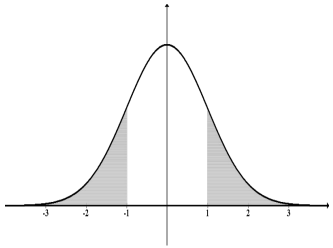
Protect privacy (at cost of utility)

Background

Federated Learning (FL) + **Secure Aggregation (SecAgg)** + **Differential Privacy (DP)**







Injected distributively, i.e., distributed DP

Train high utility models

Security at the cost of increased communication

Protect privacy (at cost of utility)

= Utility-Communication-Privacy Tradeoff

Setup & Main Approach

- with ℓ_2 bounded $d-$dimensional vectors $\|x_i\|_2 < c, i \in [n]$
- DP DME: $\min \|S \cdot \hat{\mu}(S^T x_1, \ldots, S^T x_n) - \mu(x_1, \ldots, x_n)\|_2$

  subject to $(\varepsilon, \delta) - DP$ where $\mu(x_1, \ldots, x_n) := \frac{1}{n} \sum_i x_i$

  with secure aggregation

Setup & Main Approach

- with ℓ_2 bounded $d-$dimensional vectors $\|x_i\|_2 < c, i \in [n]$
- DP DME: $\min \|S \cdot \hat{\mu}(S^T x_1, \ldots, S^T x_n) - \mu(x_1, \ldots, x_n)\|_2$

  subject to $(\varepsilon, \delta) - DP$ where $\mu(x_1, \ldots, x_n) := \frac{1}{n} \sum_i x_i$

  with secure aggregation

Compress x_i

- construct sparse random projection matrix $S^T :=$

$$\overbrace{\begin{bmatrix} 0 & -1 & 0 & \cdots & 0 \\ 1 & 0 & -1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix}}^{d} \Bigg\} m = t \times w$$

random one-hot vector in $\mathbb{R}^m$ with random sign

- perform $\hat{\mu}(x_1, \ldots, x_n) = S \cdot \frac{1}{n} \sum_i \big((S^T x_i) + \mathcal{N}(0, z \cdot c)\big)$

server | client

Setup & Main Approach

- with ℓ_2 bounded $d-$dimensional vectors $\|x_i\|_2 < c, i \in [n]$
- DP DME: $\min \|S \cdot \hat{\mu}(S^T x_1, \ldots, S^T x_n) - \mu(x_1, \ldots, x_n)\|_2$

  subject to $(\varepsilon, \delta) - DP$ where $\mu(x_1, \ldots, x_n) := \frac{1}{n} \sum_i x_i$

  with secure aggregation

Compress x_i

- construct sparse random projection matrix $S^T :=$

$$\overbrace{\begin{bmatrix} 0 & -1 & 0 & \cdots & 0 \\ 1 & 0 & -1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix}}^{d} \Bigg\} m = t \times w$$

random one-hot vector in $\mathbb{R}^m$ with random sign

- perform $\hat{\mu}(x_1, \ldots, x_n) = S \cdot \frac{1}{n} \sum_i \big((S^T x_i) + \mathcal{N}(0, z \cdot c)\big)$

server | client

We must use linear encoding schemes due to secure aggregation

Main Theoretical Results

$O(min(n^2\varepsilon^2 \log n, d))$ bits are both necessary and achievable and, achieves optimal MSE as $O\left(\frac{c^2 d}{n^2\varepsilon^2}\right)$ under $(\varepsilon, \delta) - DP$.

- Communication decreases with more privacy!

Theorem 5.2

Main Theoretical Results

$O(min(n^2\varepsilon^2 \log n, d))$ bits are both necessary and achievable and, achieves optimal MSE as $O\left(\frac{c^2 d}{n^2\varepsilon^2}\right)$ under $(\varepsilon, \delta) - DP$.

- Communication decreases with more privacy!

Theorem 5.2

Assume $\|\mu(x_1, \ldots, x_n)\|_0 \leq s$, then only $O\left(s \log d \log\left(n^2 + s \log\left(d/\varepsilon^2\right)\right)\right)$ bits are needed for $O\left(\frac{c^2 s \log^2 d}{n^2\varepsilon^2}\right)$ MSE.

Theorem 6.1

- If the sum is sparse, we can compress more and obtain lower MSE!
- How? By using compressed sensing and LASSO decoding.

Main Empirical Results

- Stack Overflow Next Word Prediction (SONWP) has *N=342,477* clients. We use *b=18*.
- Federated EMNIST (F-EMNIST) has *N=3400* clients. We use *b=16* .
- Define a relative slack, *Δ*, for accuracy-drop relative to *r=1*.
- Compare within the same noise multiplier, *z*.

Main Empirical Results

- Stack Overflow Next Word Prediction (SONWP) has *N=342,477* clients. We use *b=18*.
- Federated EMNIST (F-EMNIST) has *N=3400* clients. We use *b=16* .
- Define a relative slack, *Δ*, for accuracy-drop relative to *r=1*.
- Compare within the same noise multiplier, *z*.

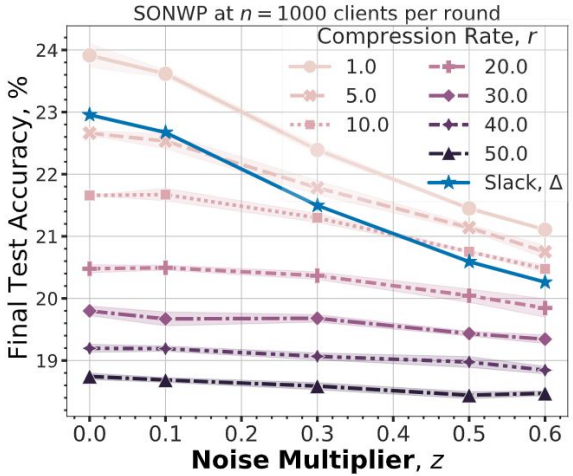


Fig. 1: ***Higher noise multiplier, z, implies higher compression.*** Δ=4%. With *z*=0.5, we get r=10x. Even without DP we get about *3x*.

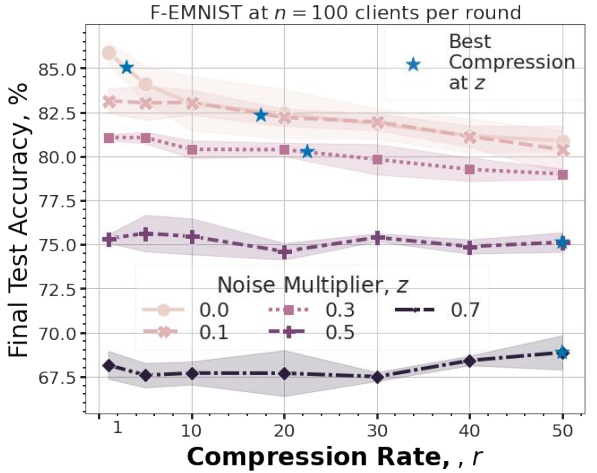


Fig. 2: ***Higher compression implies tighter privacy.*** Δ=1%. At *r=20x*, *z=0.3* can be obtained `for free'.

Main Empirical Results

- Stack Overflow Next Word Prediction (SONWP) has *N=342,477* clients. We use *b=18*.
- Federated EMNIST (F-EMNIST) has *N=3400* clients. We use *b=16* .
- Define a relative slack, *Δ*, for accuracy-drop relative to *r=1*.
- Compare within the same noise multiplier, *z*.

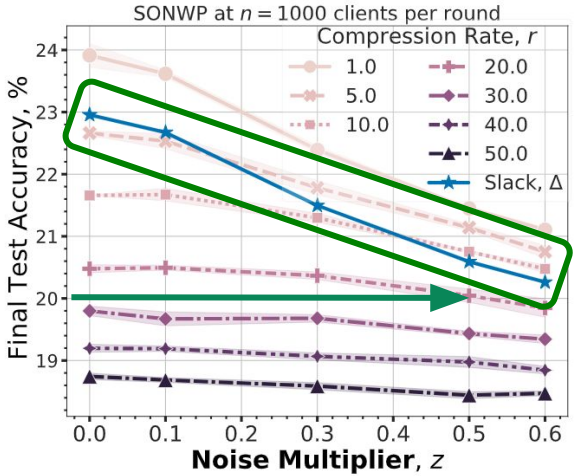


Fig. 1: ***Higher noise multiplier, z, implies higher compression.*** Δ=4%. With *z*=0.5, we get r=10x. Even without DP we get about *3x*.

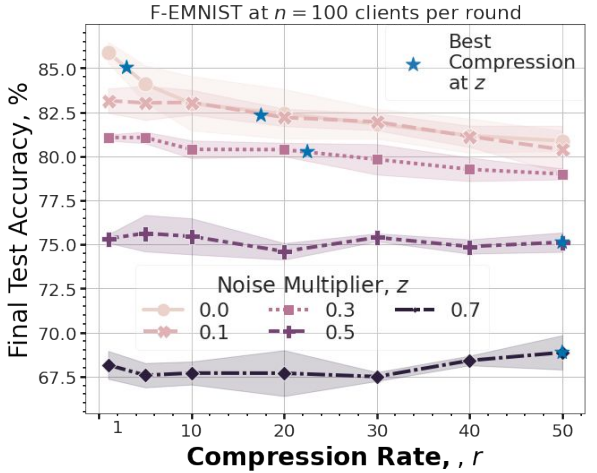


Fig. 2: ***Higher compression implies tighter privacy.*** Δ=1%. At *r=20x*, *z=0.3* can be obtained `for free'.

Sketching or (Linear) Quantization?

- Fix *z=0.5* and *n=100* for F-EMNIST
- Vary sketch compression *r* and the bit width *b*
- Previous best accuracy: ~75.25%, *@b=16, r=1x*
- Allow slack *Δ=1%*
- Cannot go lower than b=10 bits/param
- Optimizing both: *0.24* bits per param @ *b=12,r=50x*

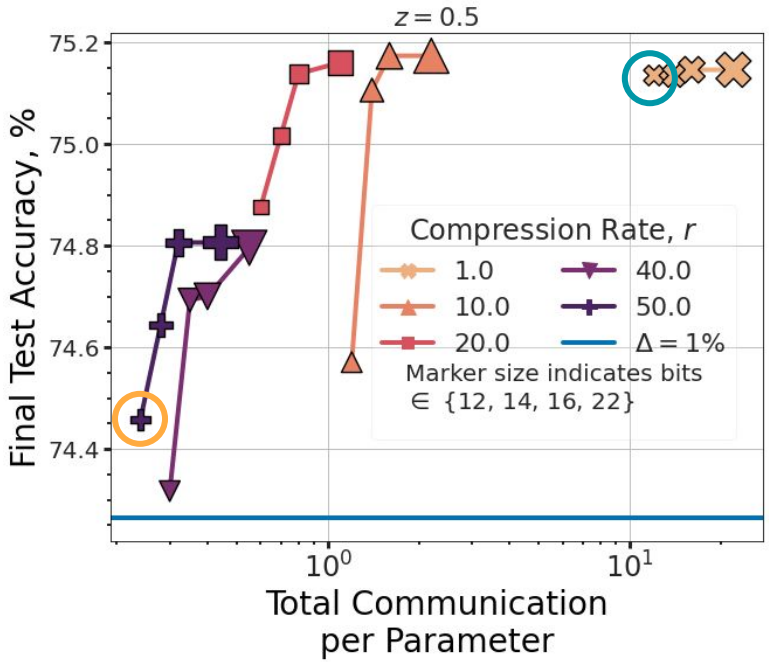


Fig. 3: Optimizing both *r* and *b* can further decrease communication, to *0.24* bits per parameter at *z= 0.5*.

Code: https://github.com/google-research/federated/tree/master/private_linear_compression

Conclusions

- Fundamental characterization of privacy-utility-communication tradeoff under secure aggregation
- Theoretical analysis well-matched by empirical results
- Practical benefits down to *0.24* bits/param on large-scale tasks
- May enable increasing the number of clients to improve utility (despite secure aggregation limitations)

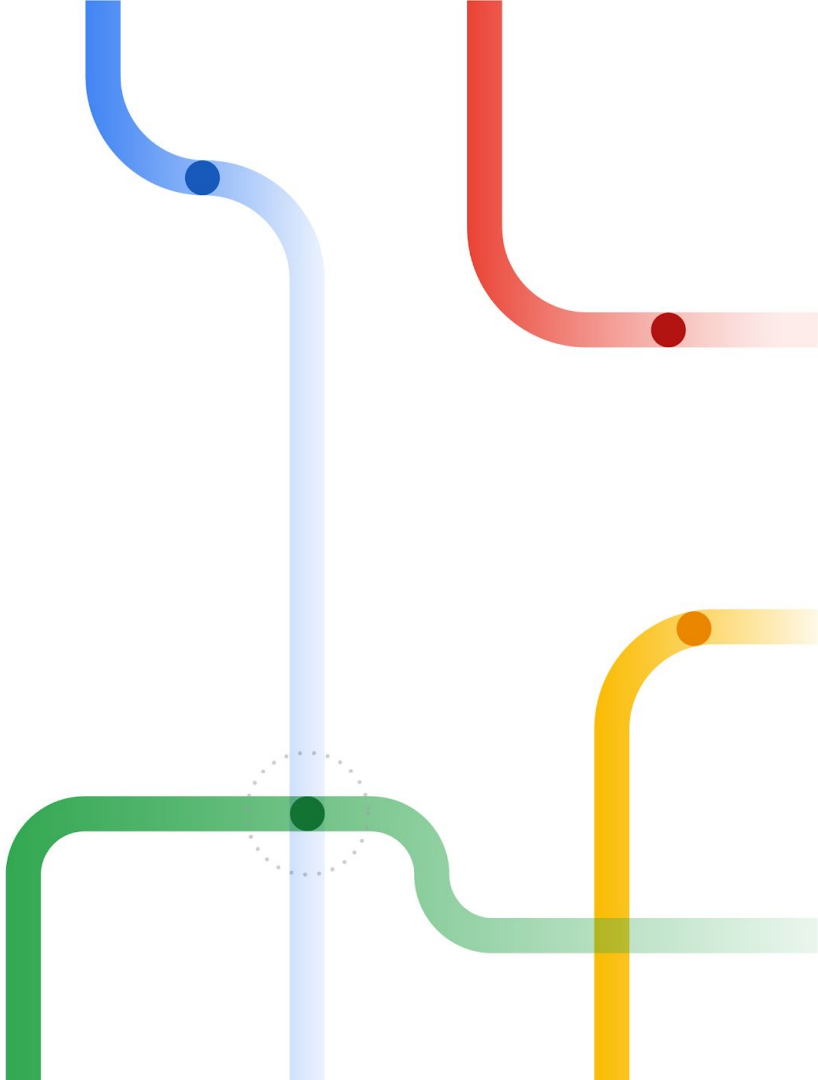
Noise Multiplier, z	Number of Clients, n	Compression Rate, r	Final Test Performance, %
0.1	100	1	83.05 ± 0.44
	1000	10	82.95 ± 0.40
0.3	100	1	80.61 ± 0.46
	1000	40	80.78 ± 0.29
0.5	100	1	75.34 ± 0.49
	1000	50	80.13 ± 0.22

Table 2. **With z sufficiently large, increasing n $= 100 \rightarrow 1000$ can attain higher model performance even for increased** r. In particular, to maintain the same SecAgg runtime, we require $r \geq 15$ for this setting to increase $n = 100 \rightarrow 1000$. We observe that $z \geq 0.3$ meets this requirement while achieving final models that outperform the $n = 100$, $r = 1$x client baseline. Results for SONWP.

Thank you for your time!

Christopher A. Choquette-Choo:
cchoquette@google.com

With: Wei-ning Chen, Peter Kairouz,
and Ananda Theertha Suresh



Google Research