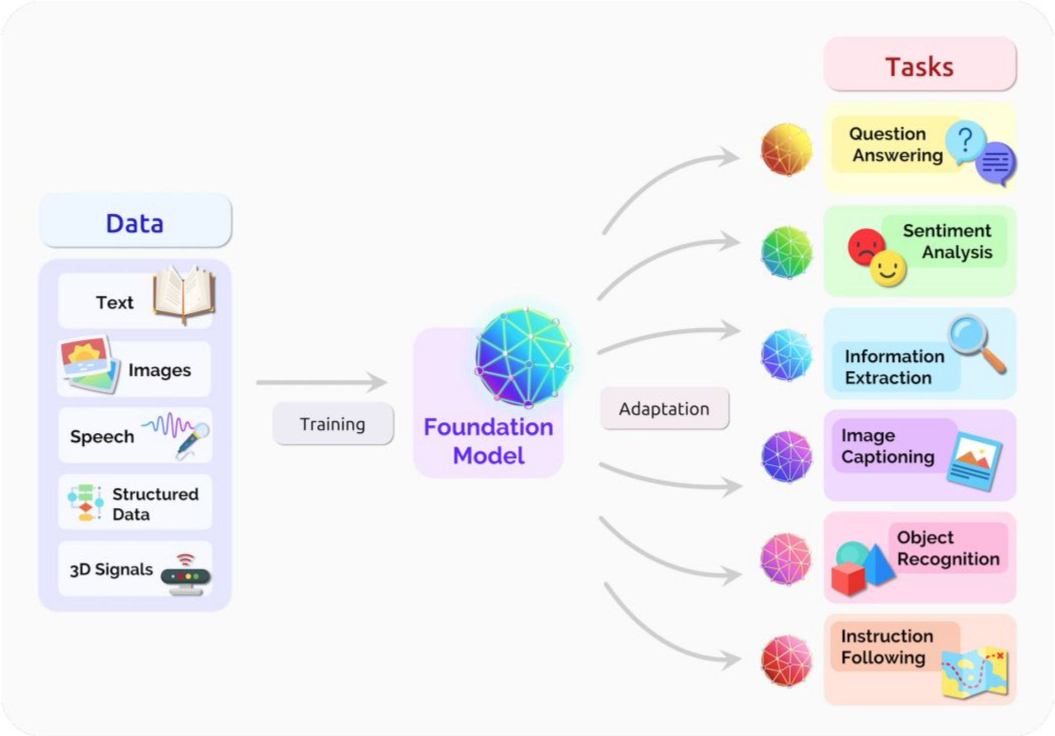
Navigating Privacy Risks in (Large) Language Models

Peter Kairouz – presenting work done with many

PPAI-24: The 5th AAAI Workshop on
Privacy-Preserving Artificial Intelligence

The rapidly evolving landscape of foundation models



Bommasani, et al. ***On the Opportunities and Risks of Foundation Models****. Stanford Center for Research on Foundation Models, Stanford Institute for Human-Centered Artificial Intelligence*

BERT [Oct '18]: Pre-text task with ~340M transformer model

GPT-3 [May '20]: Chatbot model at extreme scales (175B)

CLIP [Jan '21]: Image captioning using pre-training tricks inspired by BERT (63M)

DALL·E [Jan '21]: Text-to-image generation with a "mini" GPT-3 (12B)

LaMDA / Bard [Jan '22 / Feb '23]: Language model for dialogue applications (137B)

ChatGPT / GPT-4 [Nov '22 / March '23]: Language model for dialogue applications (175B, ~1.8T)

LLaMa / LLaMA-2 [Feb '23 / July '23]: General purpose language models (7, 13, 70B)

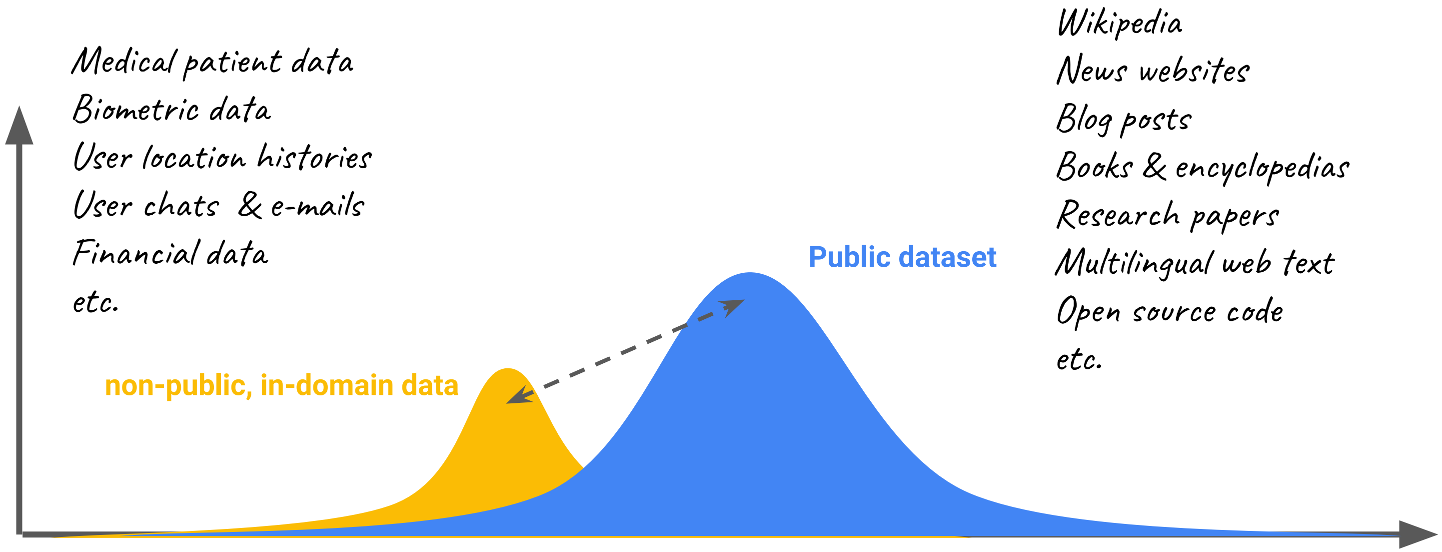
PaLM / PaLM-2 [April '22 / May '23]: Language model for dialogue applications (340B)

Gemini-1 / Gemini-1.5 / Gemma [Dec '23 / Feb '24 / Feb' 24]: A family of (natively multi-modal) foundation language models

These models are so damn good…

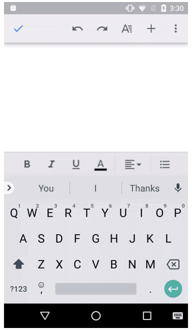
So why access non-public data?

training on data from the same distribution that we will be inferencing on ("in-domain data") gives better results



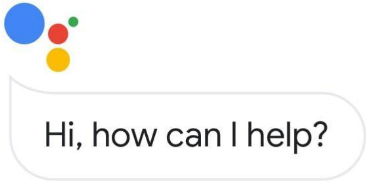
Also evidenced by Google product launches that moved training on-device:

Data minimization!



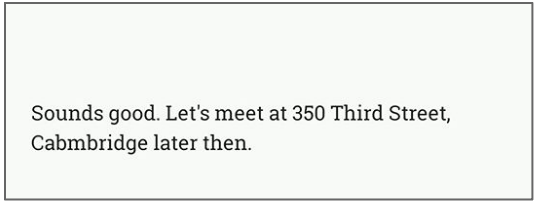
+24% accuracy
Gboard typing next-word-prediction model trained on on-device data instead of server-side logs.

Turned off server logging!



-10% hotword mis-recognition
Google Assistant hotword triggering training with on-device data that isn't sent to datacenters.

Reduced server logging!



+10% Accuracy
SmartSelect identifying long-form entities training from on-screen pixels instead of Wikipedia proxy data.

Never started server logging!

~~“LLMs don’t benefit from training on in domain data”~~

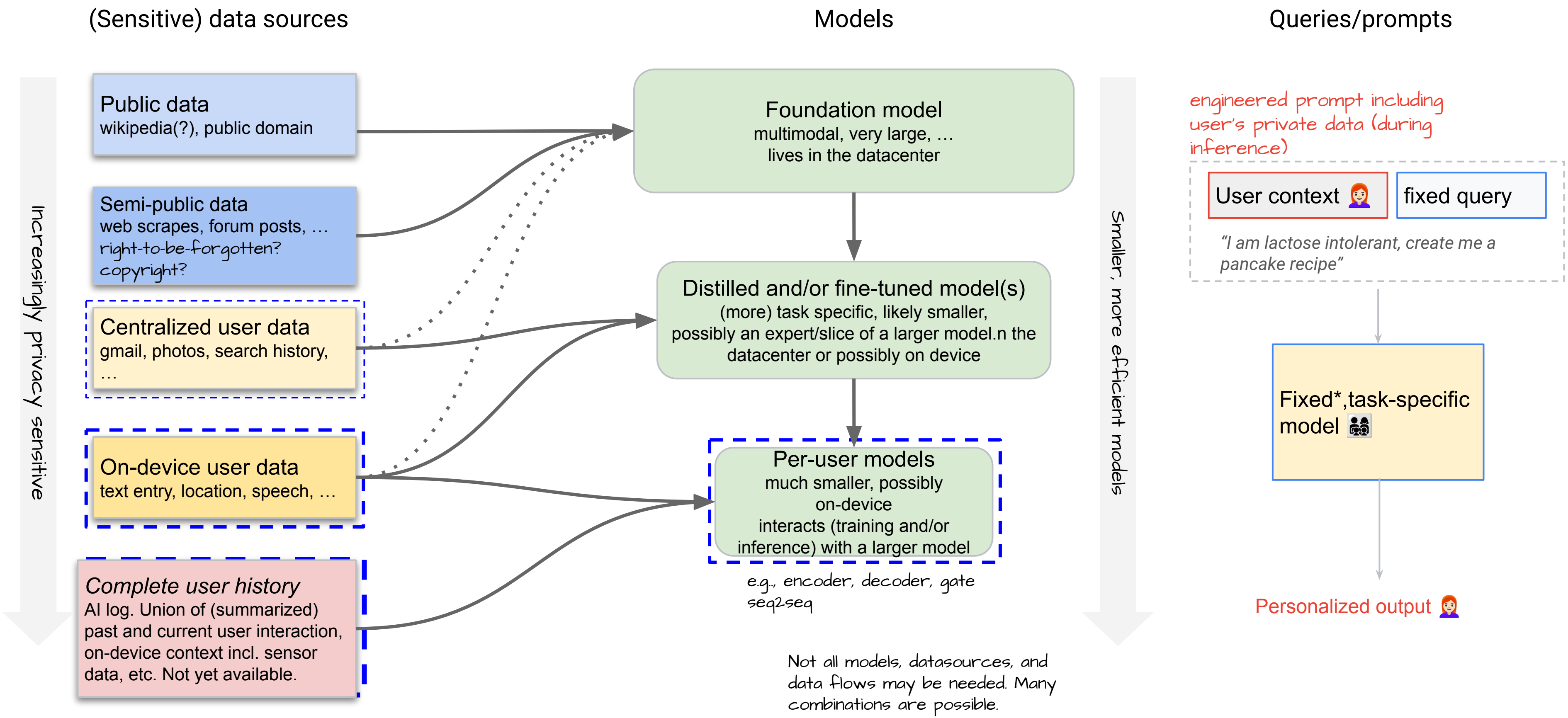
we believe:

High quality in-domain data (possibly privacy sensitive) will be required for accurate and efficient models in production.

In-domain data ⇒ need to worry about privacy!

But whose privacy are we talking about?

Whose privacy are we talking about?



Privacy principles

More details in "***Federated Learning and Privacy***"
Communications of the ACM, April 2022

For the user

Privacy principle 1

The User has *Transparency and Use-Centric Control*

(forward-looking transparency, retrospective auditability of computation or release details, control of at least the immediate use of data, e.g. use in training.)

For the platform

Privacy principle 2

Processing encodes

Data Minimization

(security, access control, focused collection, TTLs, ...)

Privacy principle 3

Released outputs provide

Data Anonymization

(differential privacy (DP), memorization auditing, ...)

For the verifiers

Privacy principle 4

Privacy claims are *verifiable*

ideally by the users themselves, by external auditors, and the service provider

Privacy principles

More details in "***Federated Learning and Privacy***"
Communications of the ACM, April 2022

For the user

The User has ***Transparency, Auditability, and Control***
of what data is used, what purpose it is used for, and how it is processed.
(forward-looking transparency, retrospective auditability of computation or release details, control of at least the immediate use of data, e.g. use in training.)

For the platform

Processing encodes
Data Minimization
(security, access control, focused collection, TTLs, …)

Released outputs provide
Data Anonymization
(differential privacy (DP), memorization auditing, …)

For the verifiers

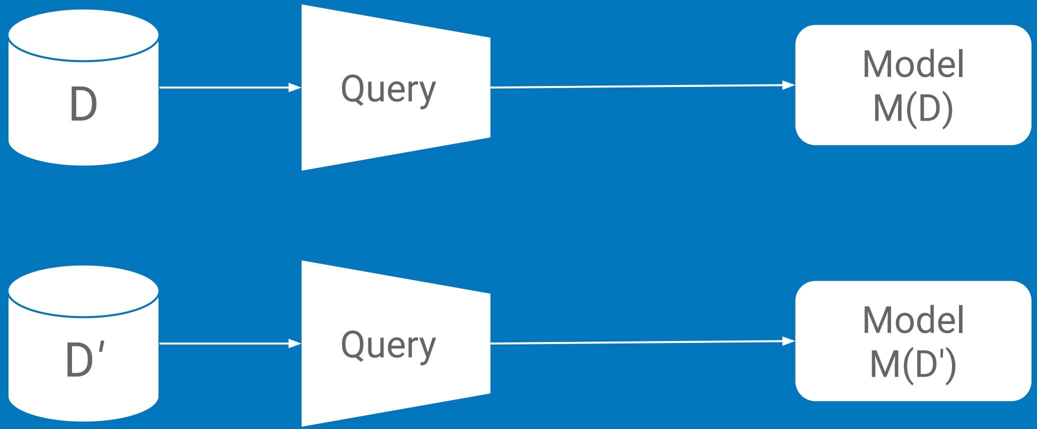
Privacy claims are ***Verifiable***
ideally by the users themselves, by external auditors, and the service provider

Differential Privacy

For ML:
Randomized training algorithm.

*When you change one **X** in the training data, the distribution of output models hardly changes*

(changes by a quantifiably small amount).



(ε, δ)-Differential Privacy: The distribution of the output $M(D)$ on database D is nearly the same as $M(D')$ for all ***adjacent databases D and D′ (differ by one unit X)***

$$\forall S: \quad \Pr[M(D) \in S] \leq \exp(\varepsilon) \cdot \Pr[M(D') \in S] + \delta$$

Example: units of privacy for language models

*When you change one **X** in the training data, the distribution of output models hardly changes. What is **X?***

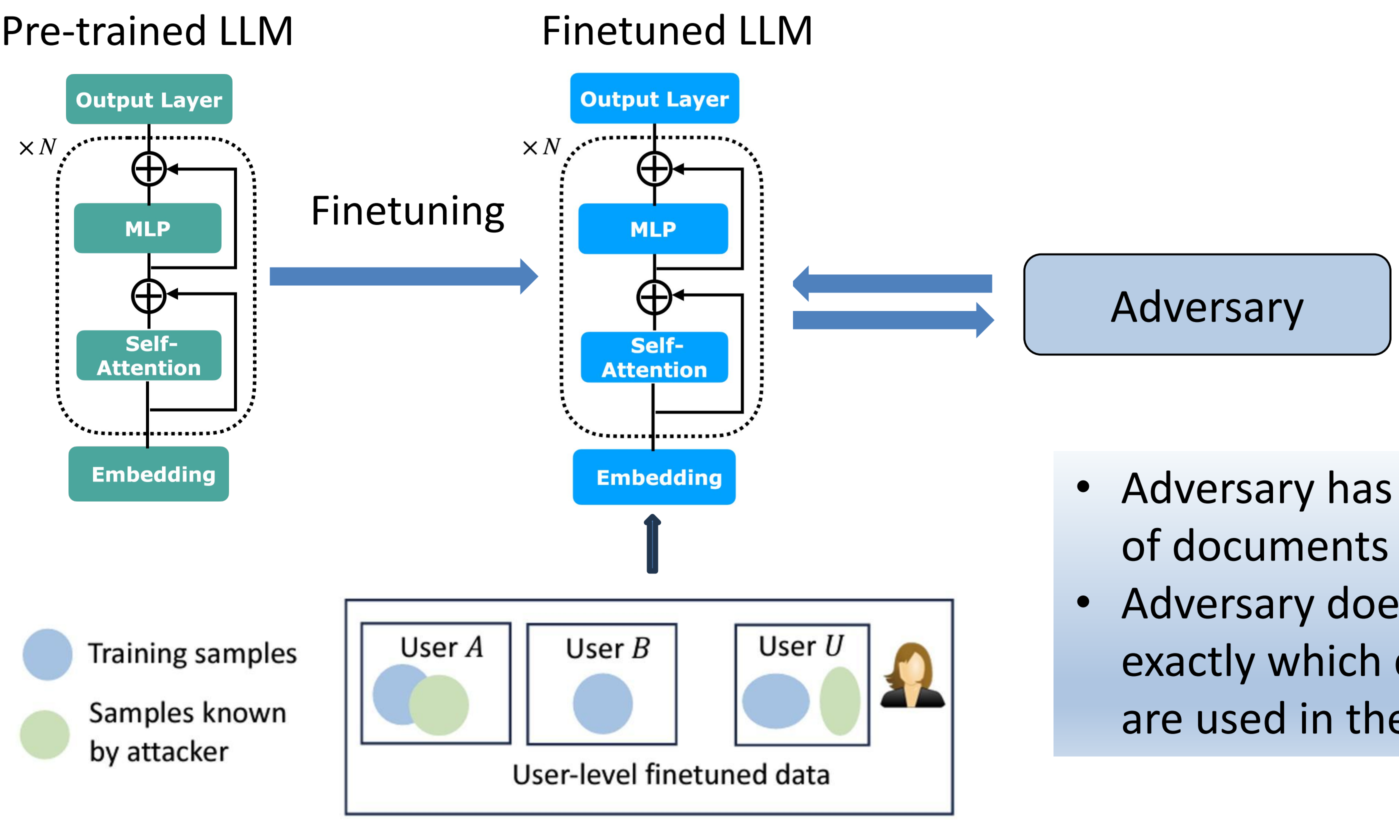
- **token**-level DP (character, wordpiece, word)
- **example**-level DP (sequence of tokens that form a row in a batch)
- **paragraph**-level DP
- **document**-level DP
- **user**-level DP
- **organization**-level DP

(Exponentially) stronger privacy at fixed $\boldsymbol{\varepsilon}$

Closest to **standard ML infra and algorithms**, but still not a perfect fit.

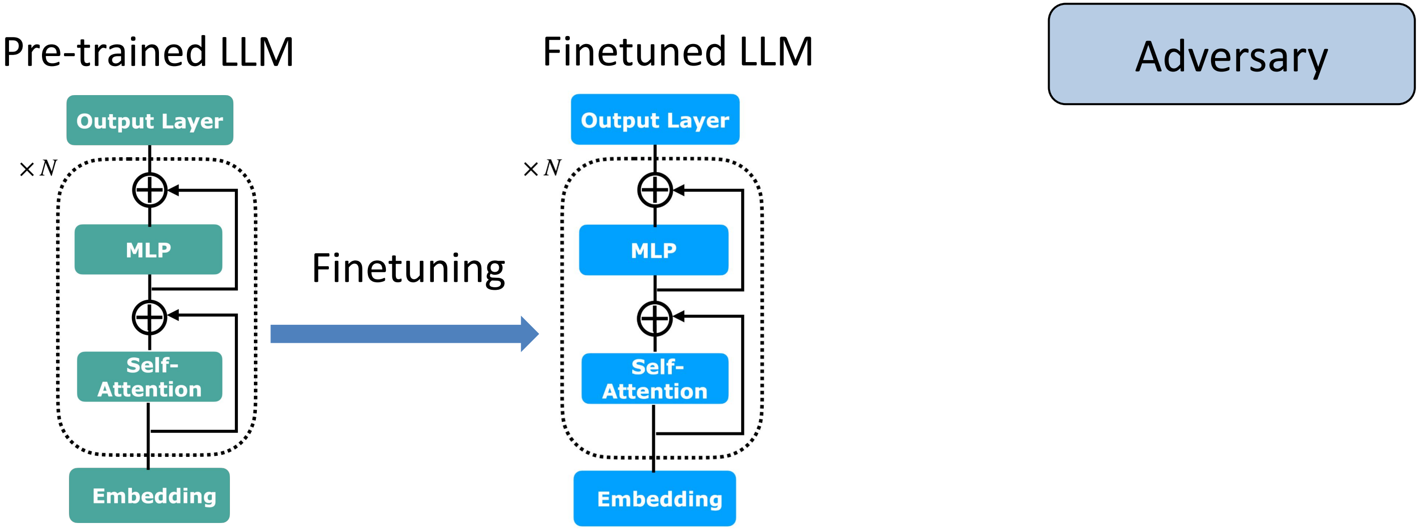
"Multi-example" notions of privacy, supported by **federated learning algorithms**. We focus on user-level, but the techniques apply to all these notions (incl. example-level).

User inference: attacker knowledge

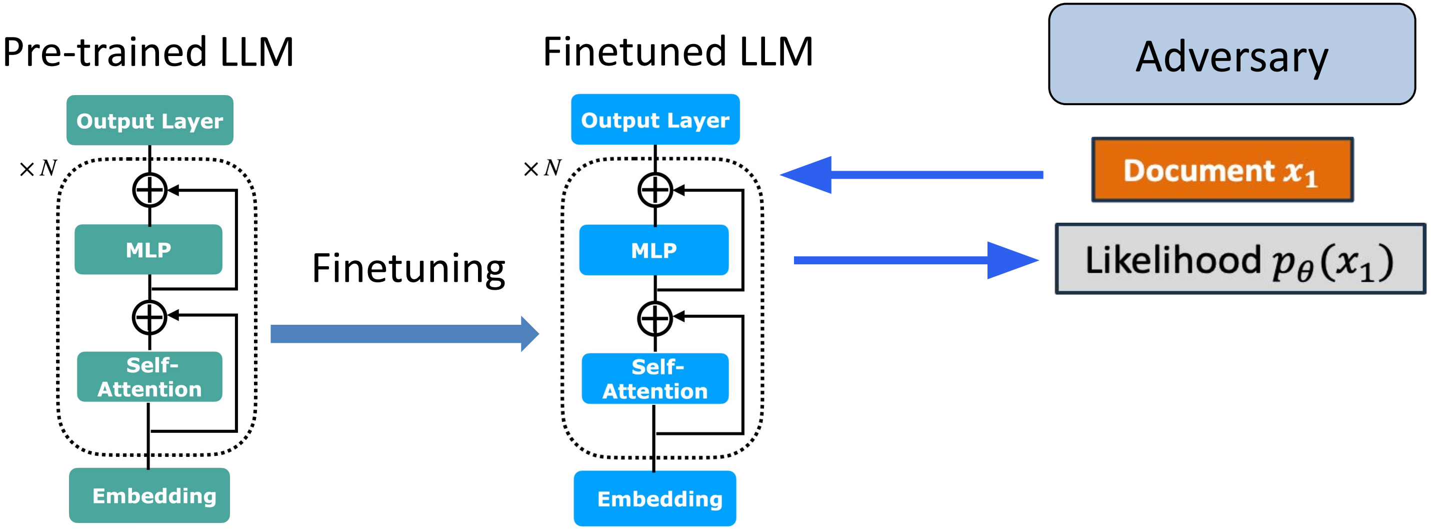


- Adversary has access to a subset of documents by target user
- Adversary does not know exactly which documents (if any) are used in the finetuning set

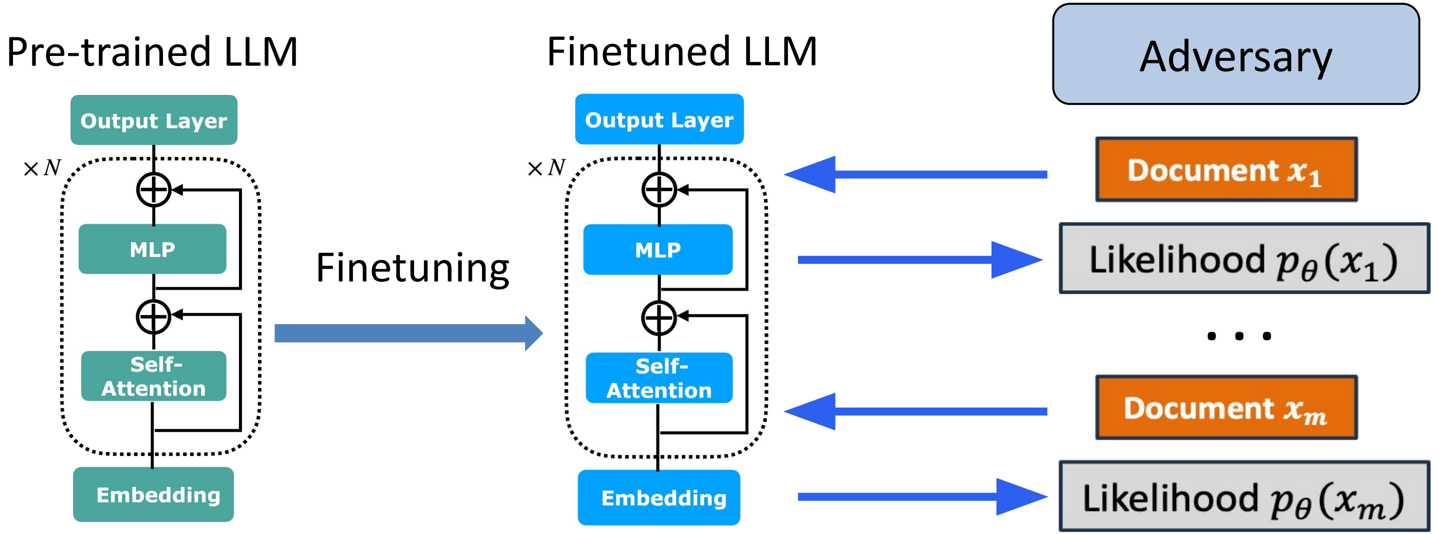
User inference attack



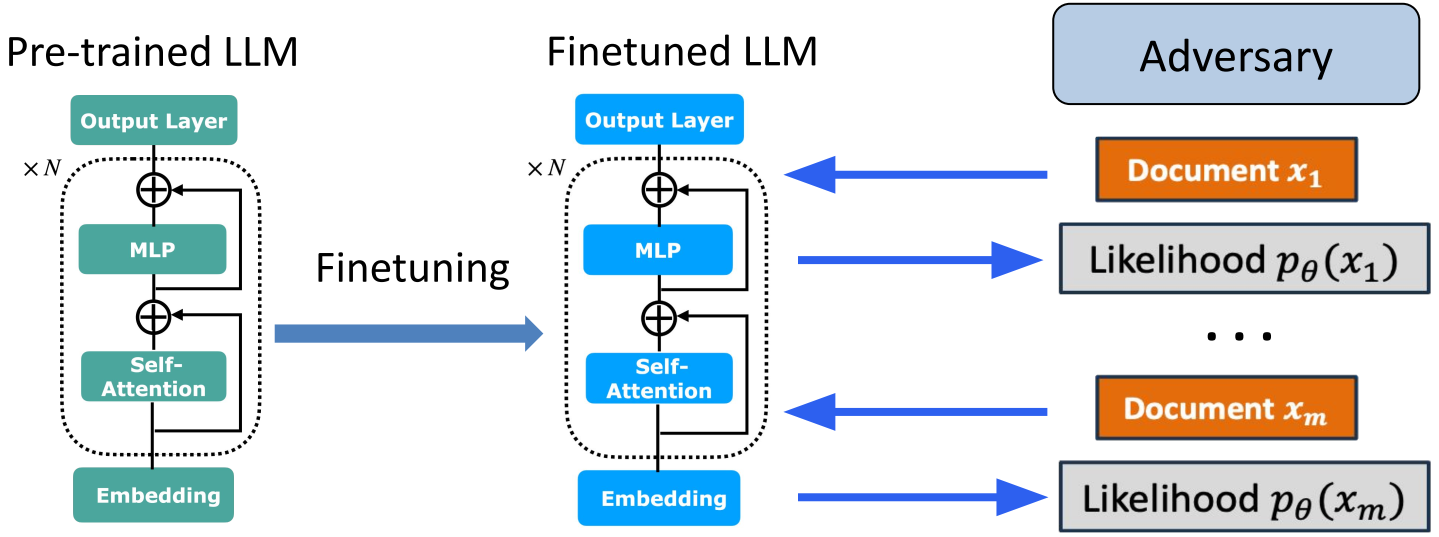
User inference attack



User inference attack

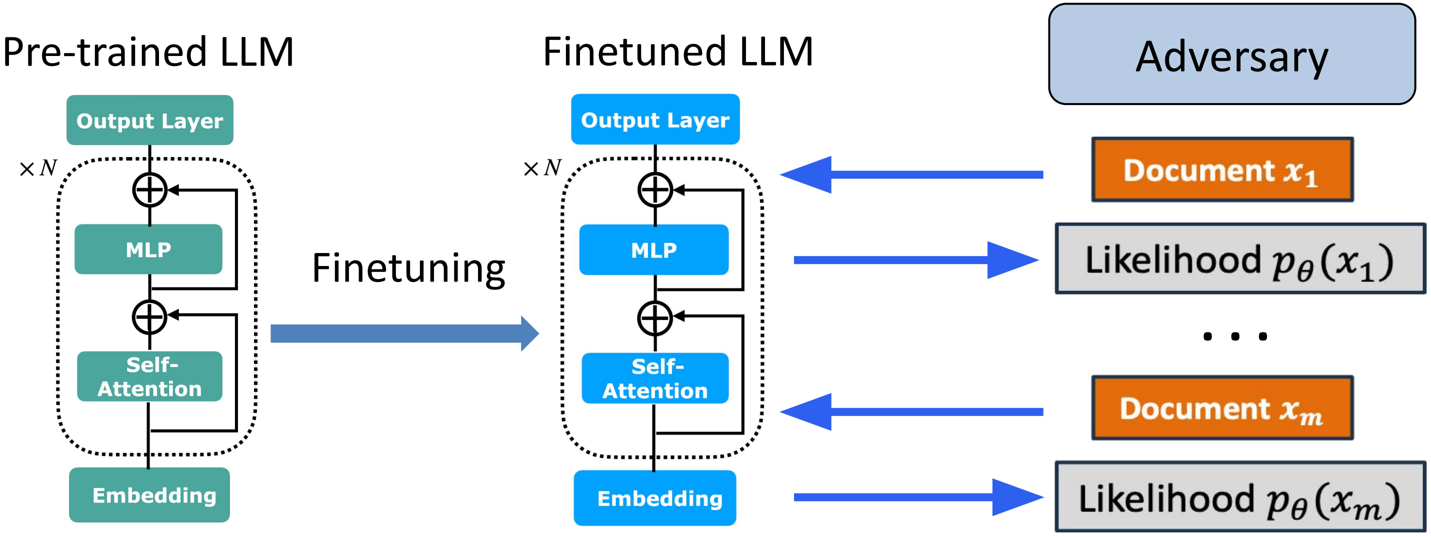


User inference attack



If average log likelihood $\frac{1}{m}\sum_{i=1}^{m}\log p_\theta(x_i)$
is high, user was in finetuning

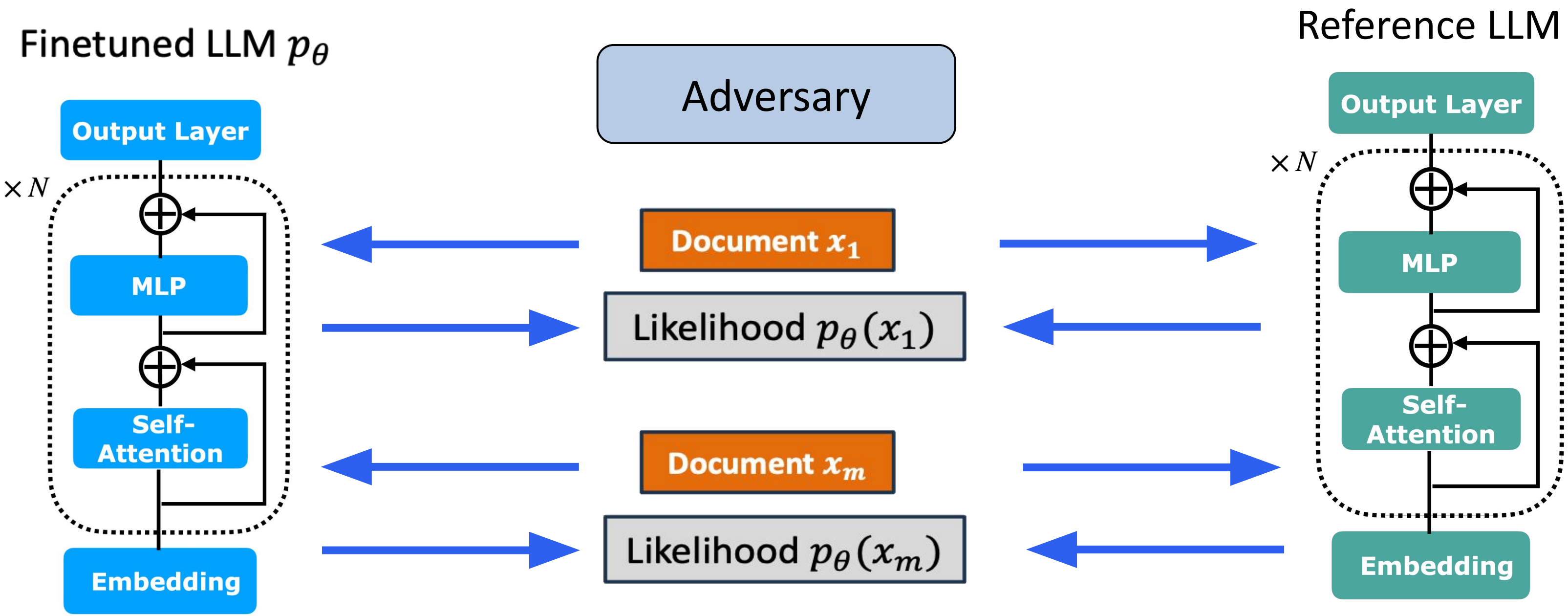
User inference attack



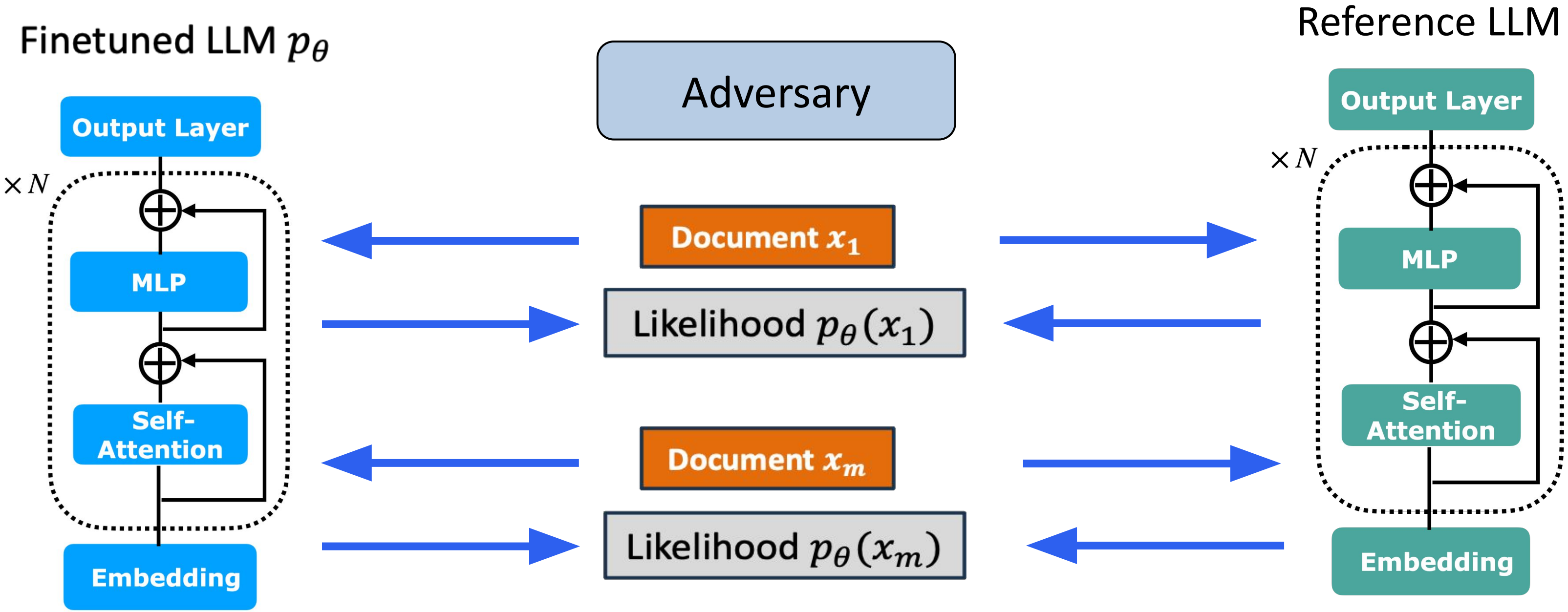
If average log likelihood $\frac{1}{m}\sum_{i=1}^{m} \log p_\theta(x_i)$
is high, user was in finetuning

Will suffer from high false positives because it's possible some sequences are "easy to predict" (i.e. appear elsewhere in the wild)

Calibrated user inference attack



Calibrated user inference attack

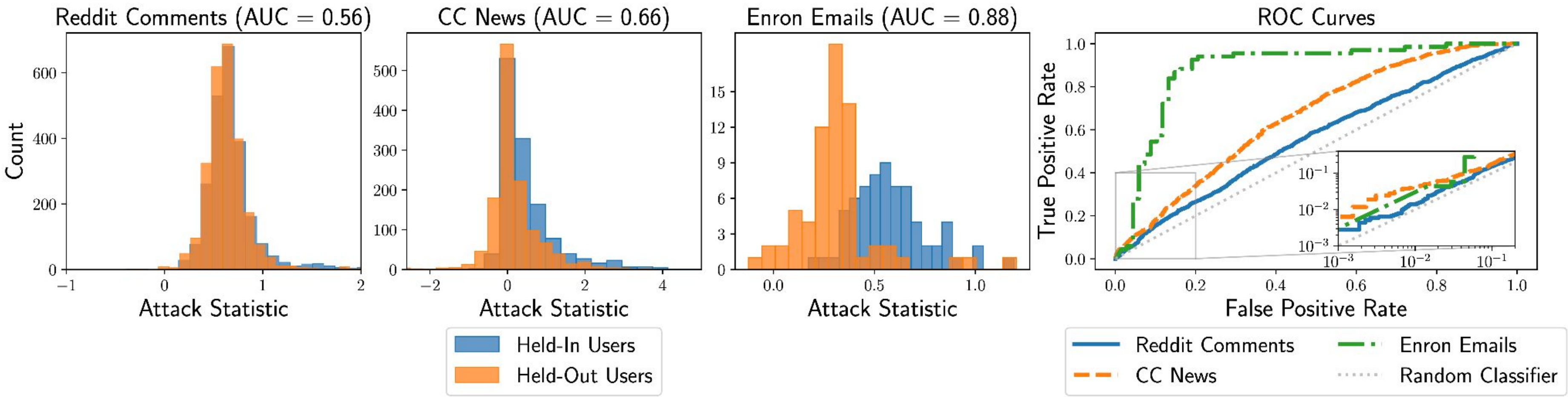


Compute **calibrated** average log likelihood

$$\mathrm{T}(\mathrm{x}_1, \ldots, x_m) = \frac{1}{m} \sum_{i=1}^{m} \log \frac{p_\theta(x_i)}{p_{\mathrm{ref}}(x_i)}$$

User U was in finetuning if $\mathrm{T}(\mathrm{x}_1, \ldots, x_m) > \tau$

Attack success on different datasets



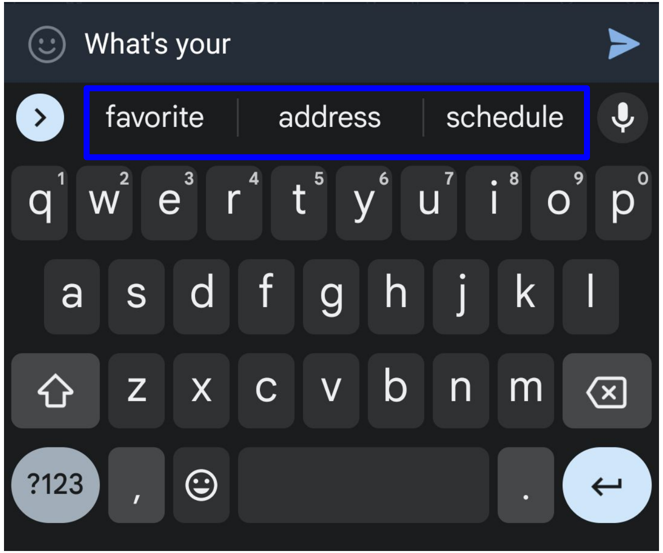
User Inference Attacks on Large Language Models, Kandpal et al., arxiv: 2310.09266

This demonstrates the importance of training with user-level DP!

And that’s exactly what we have been doing for years with Gboard..

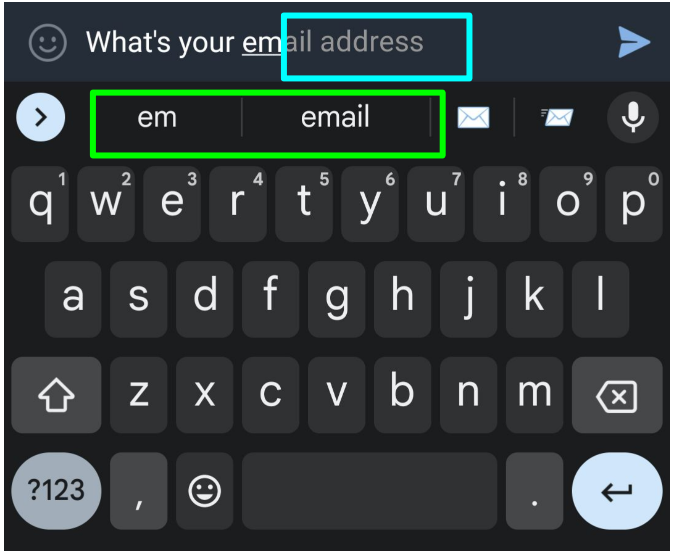
Case study: Gboard language models

Gboard Next Word Prediction (NWP)



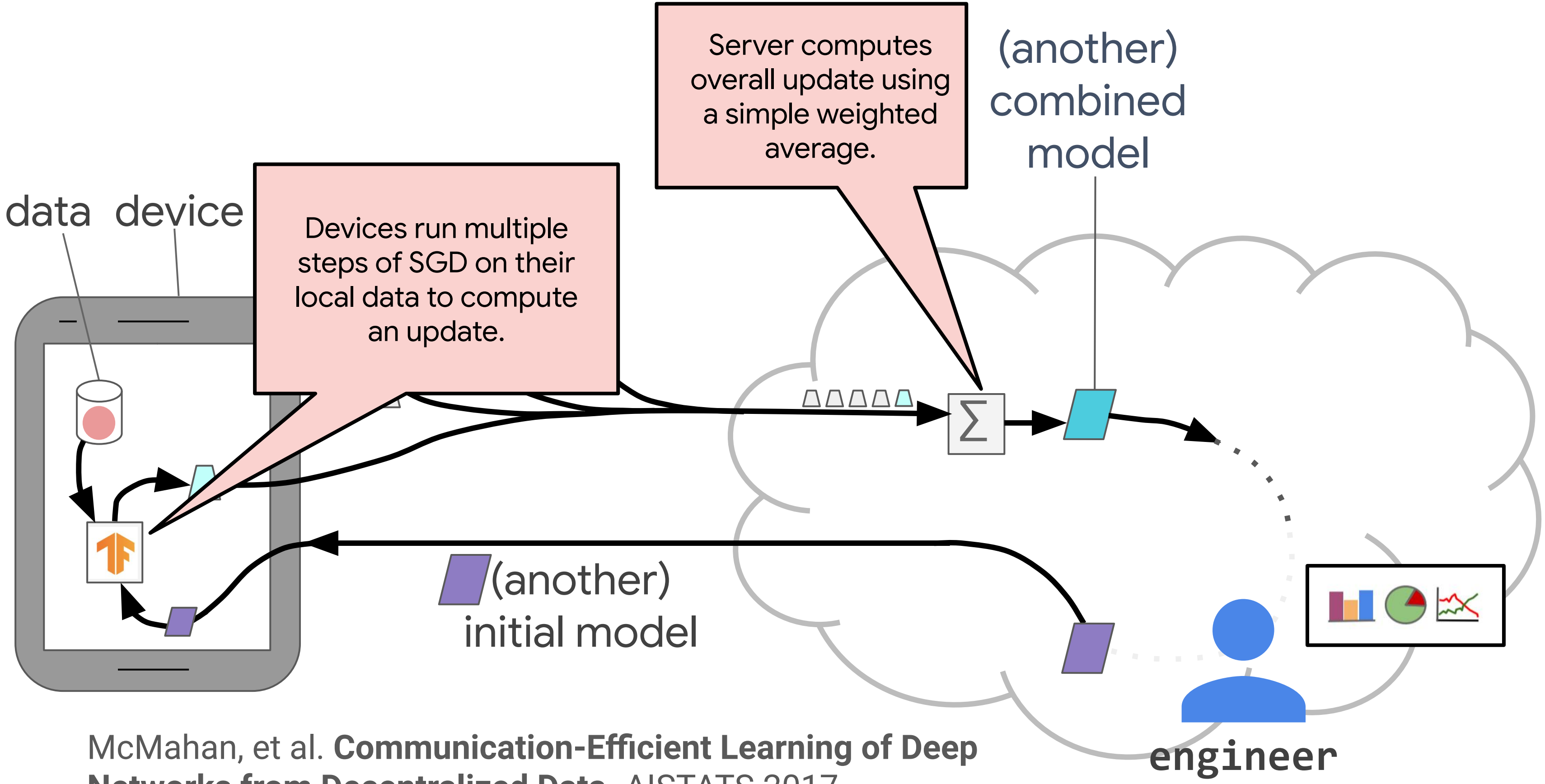
NWP LM: ~2.4M / 4.4M parameters

Smart Compose (SC)
On-The-Fly Rescoring(OTF)



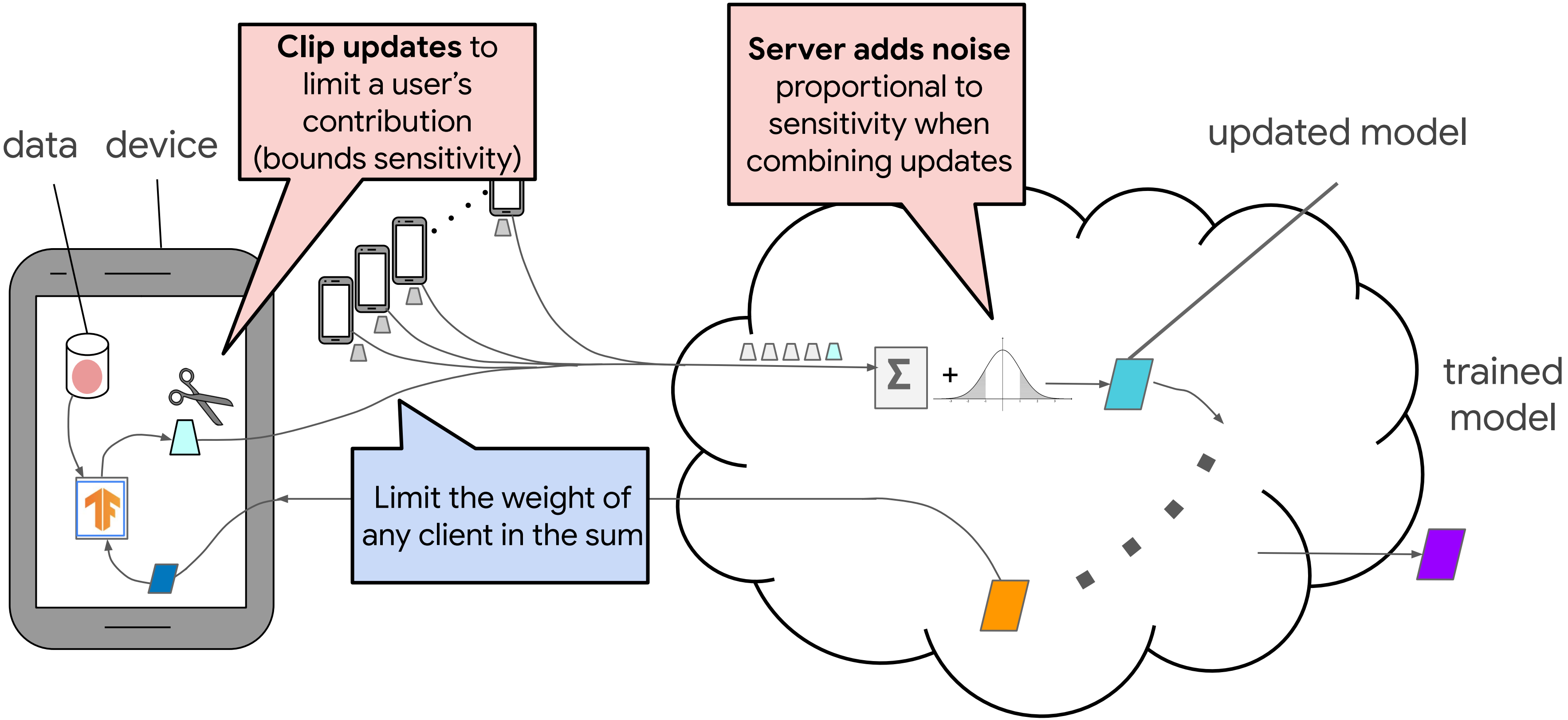
OTF LM: ~6.4M parameters

Federated Averaging (FedAvg) algorithm



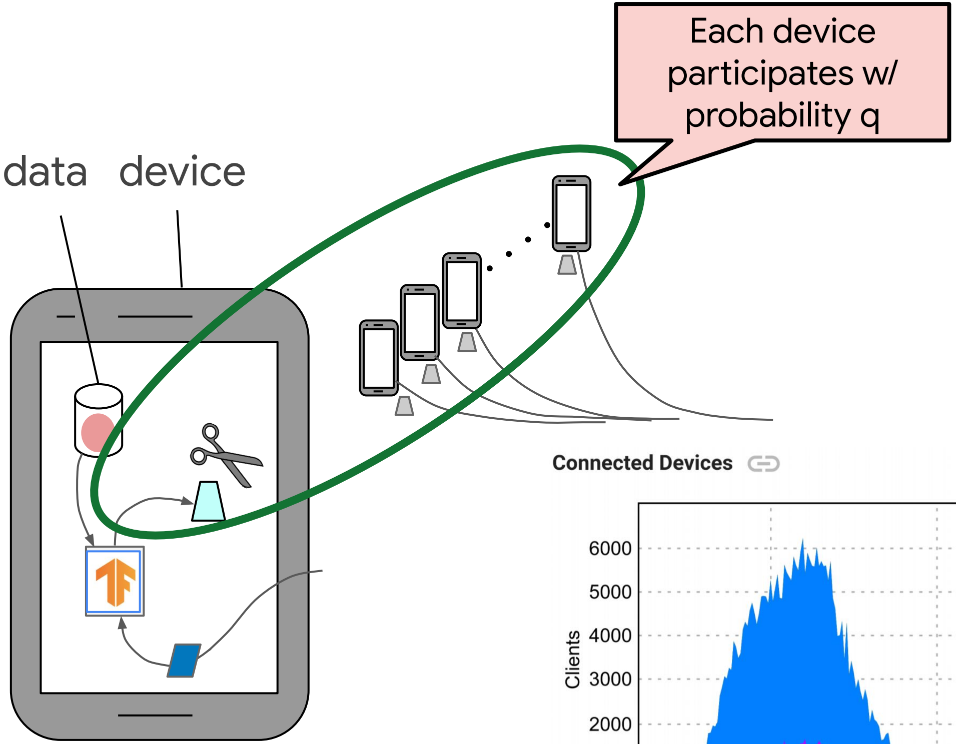
McMahan, et al. **Communication-Efficient Learning of Deep Networks from Decentralized Data.** AISTATS 2017.

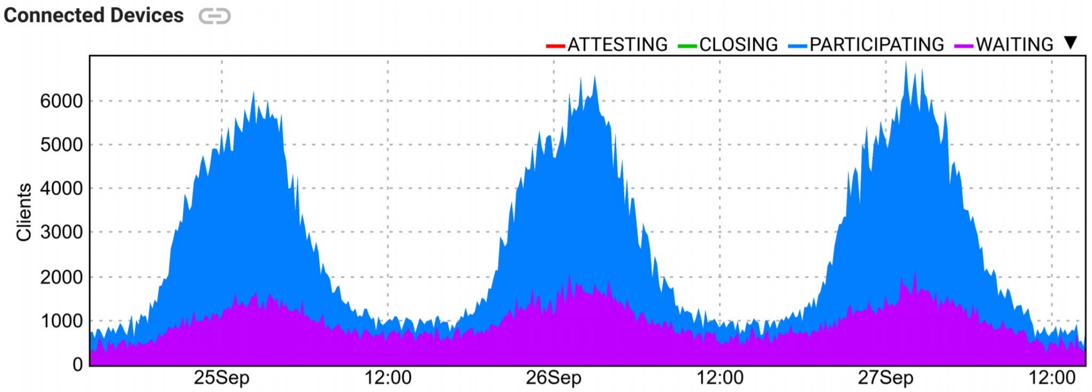
Differentially Private Federated Averaging (DP-FedAvg)



Mcmahan, Ramage, Talwar, Zhang. ***Learning Differentially Private Recurrent Language Models.*** *ICLR 2018*

Differentially Private Federated Averaging (DP-FedAvg)

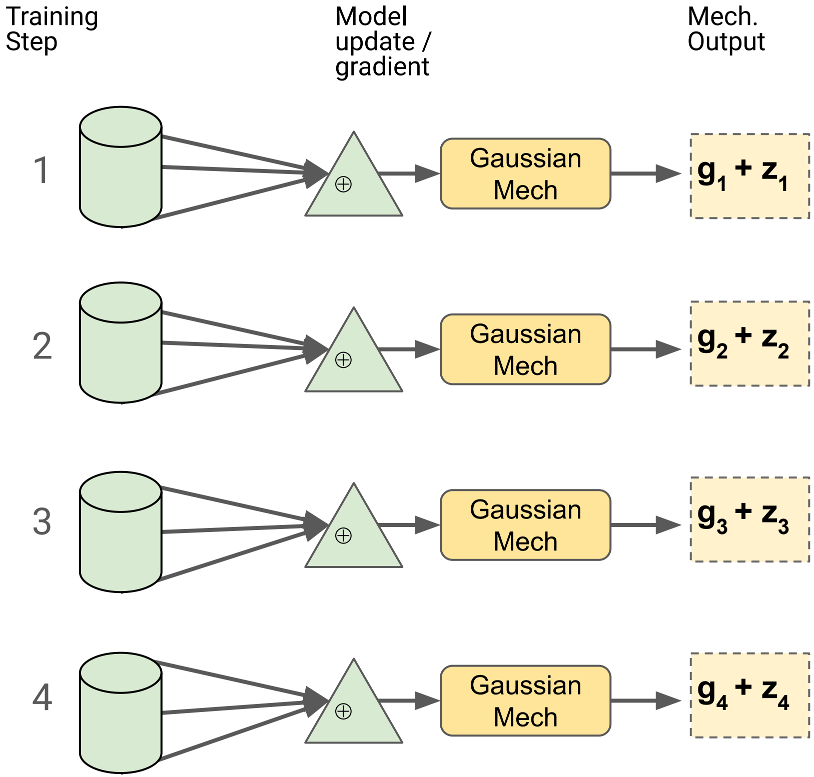




Challenges

- Hard or even impossible to uniformly sample clients from underlying population
- Need to preserve privacy and make progress even as the set of devices available varies with time arbitrarily
- For efficiency, clients should decide locally when to connect to the server

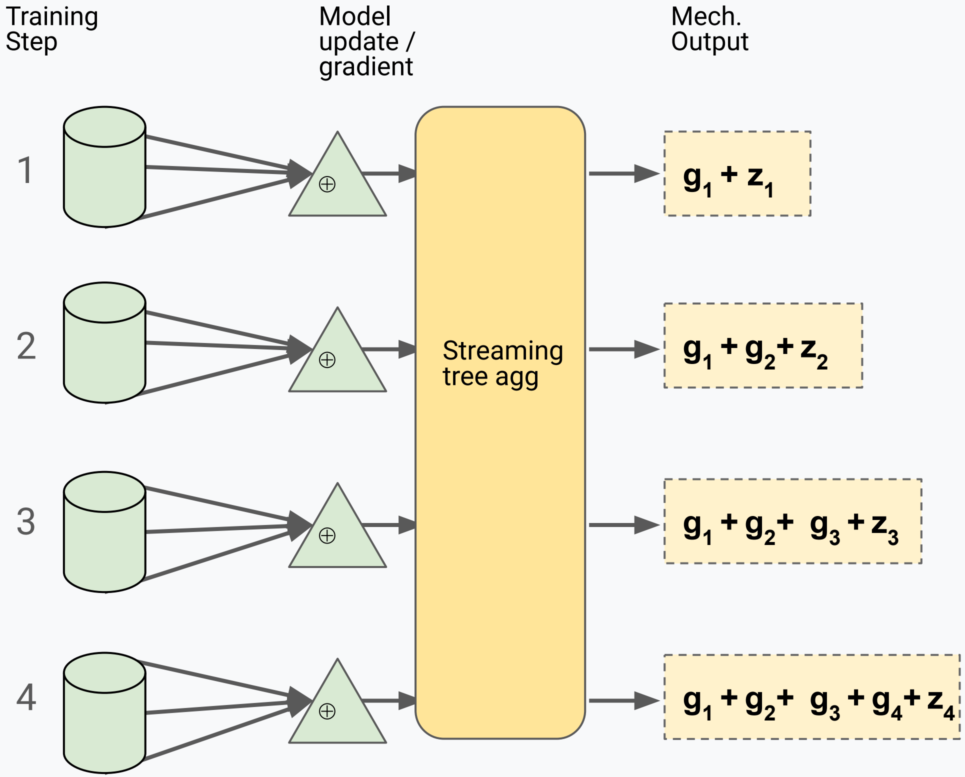
DP FedAvg



- **Independent noise** is added to each round
- Relies heavily on **amplification-by-sampling**

DP-FTRL

Kairouz, McMahan, Song, Thakkar, Thakurta, Xu. ***Practical and Private (Deep) Learning without Sampling or Shuffling.*** *ICML 2021*

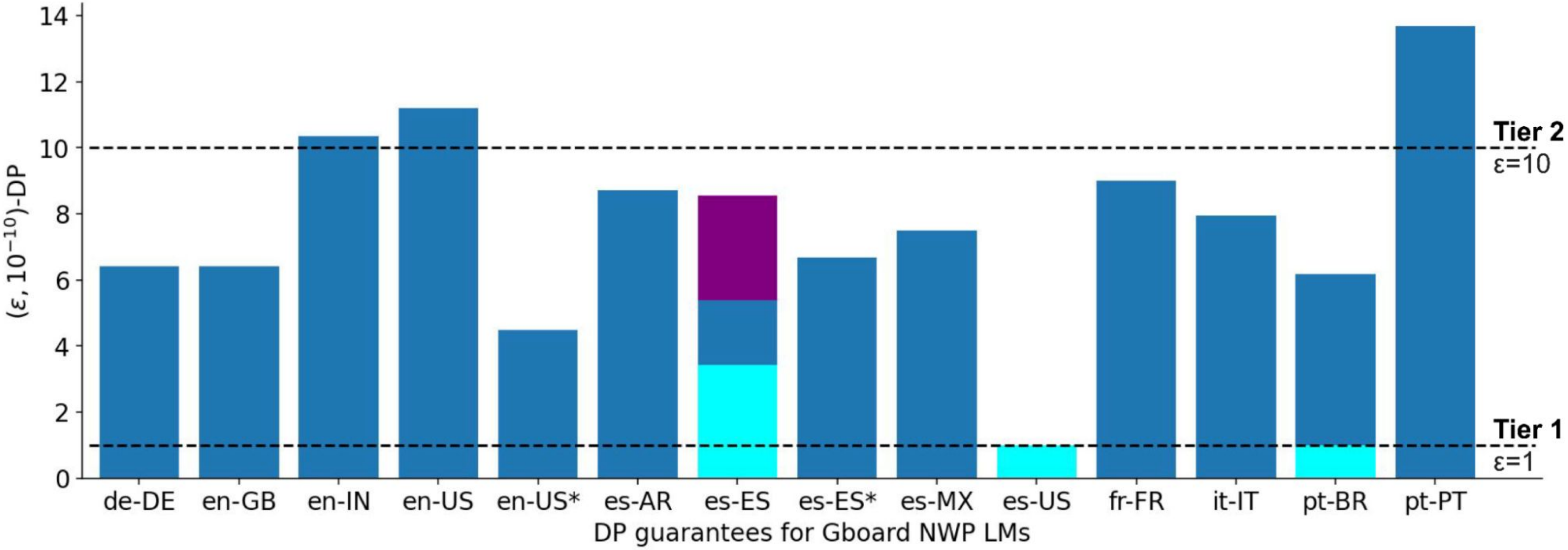


- **Correlated noise** is added in each round
- Competitive with DP-FedAvg w/ amplification.

"All the next word prediction neural network LMs in Gboard now have DP guarantees, and all future launches of Gboard neural network LMs will require DP guarantees."

Federated Learning of Gboard Language Models with Differential Privacy, June 2023

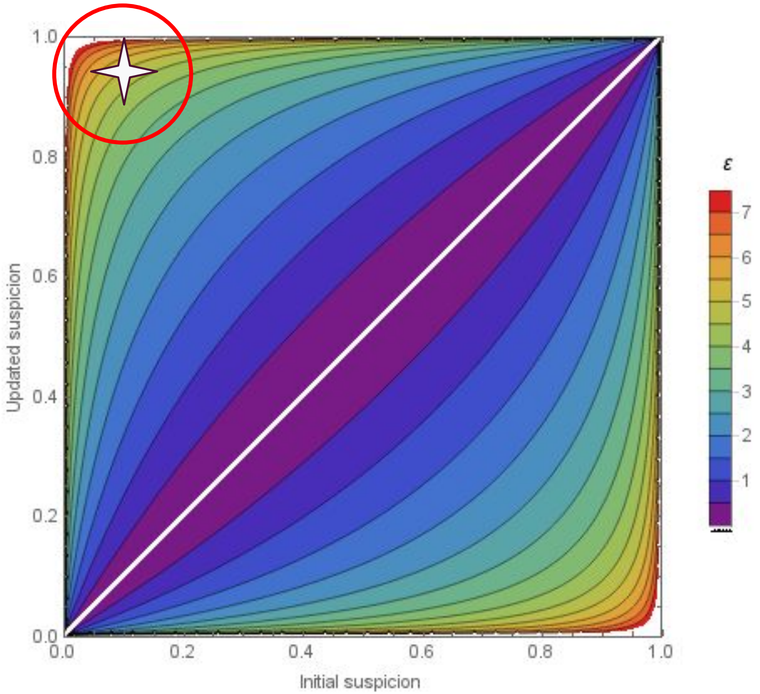
Strong DP guarantees with eps < 1!



DP guarantees for Gboard NWP LMs (the purple bar represents the first es-ES launch of ε=8.9; cyan bars represent privacy improvements for models trained with MF-DP-FTRL; tiers are from "How to DP-fy ML" guide; en-US and es-ES* are additionally trained with SecAgg).*

Is it okay to train with large-ish epsilons?

publication/application	ε
U.S. 2020 Census	19.6
High-accuracy image classification (De et al., 2022)	8
FL training of GBoard language models (Xu et al., 2023)	0.99–13.69



For ε=5, attacker can go from a low suspicion of 10% to a very high degree of certainty (94%).

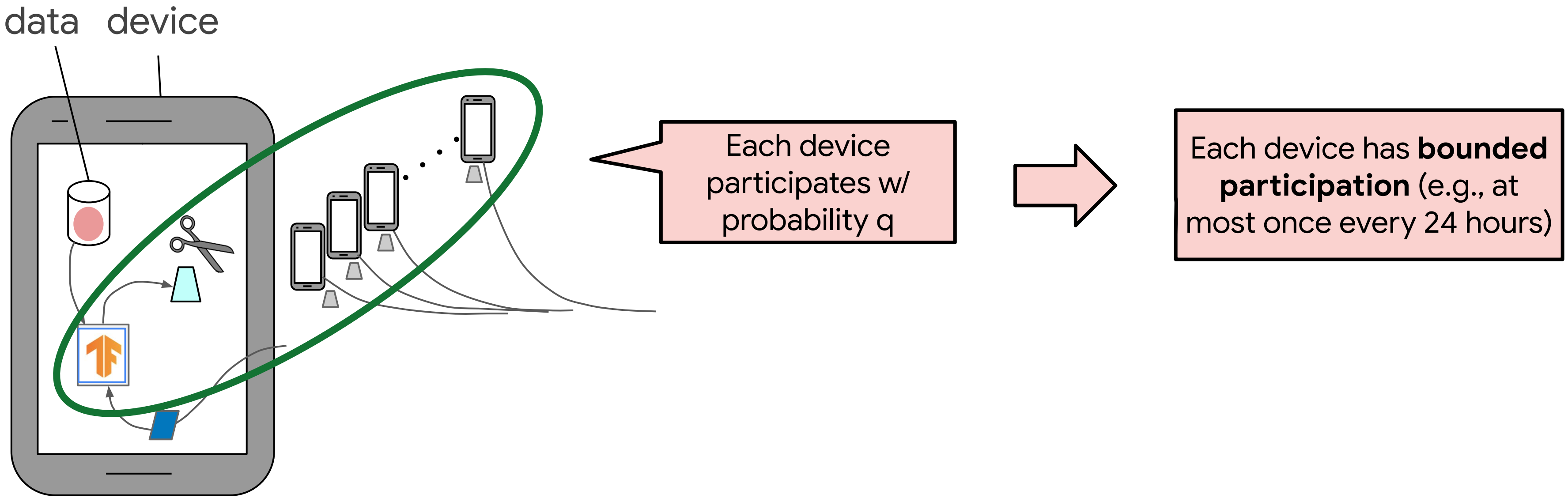
https://desfontain.es/privacy/differential-privacy-in-more-detail.html

The DP threat model assumes:

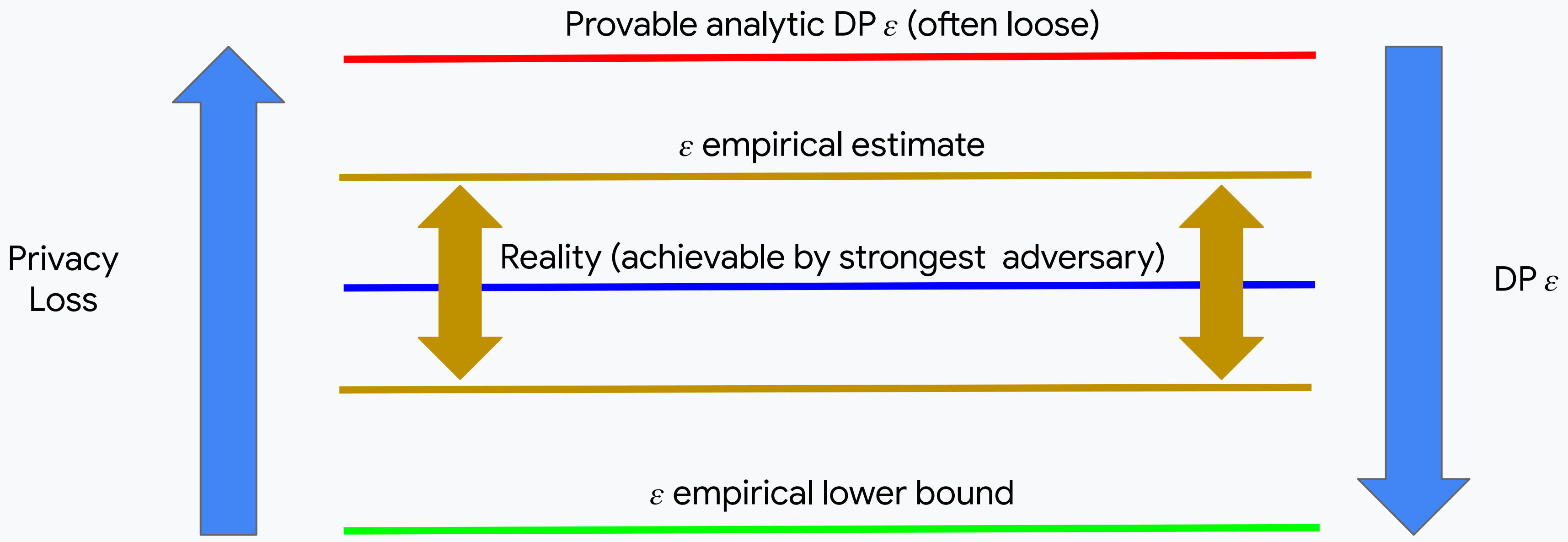
1. Worst case dataset pair ***D, D'***
2. An adversary who is trying to distinguish between ***D, D'*** (1 bit of information)
3. An infinitely powerful adversary, both computationally and statistically
4. An adversary who has (white-box) access to the model parameters
5. An adversary who sees all the model iterates in all rounds
6. Worst case participation pattern – e.g. may not take full advantage of data sampling/shuffling

The DP threat model assumes:

1. Worst case dataset pair ***D, D’***
2. An adversary who is trying to distinguish between ***D, D’*** (1 bit of information)
3. An infinitely powerful adversary, both computationally and statistically
4. An adversary who has (white-box) access to the model parameters
5. **An adversary who sees all the model iterates in all rounds**
6. **Worst case participation pattern – e.g. may not take full advantage of data sampling/shuffling**

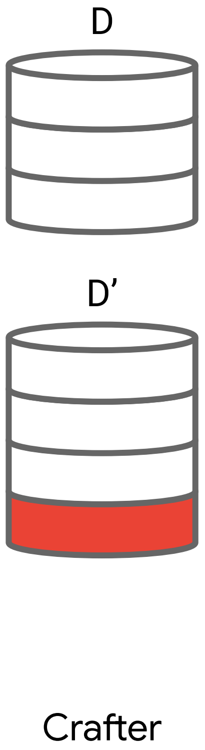


Empirical ε estimation

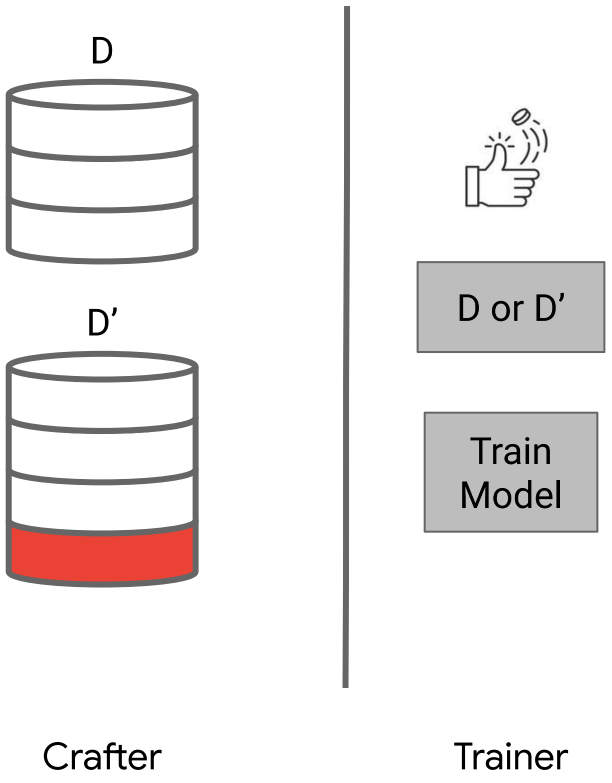


- Threat model may be too strong (e.g., release all model iterates)
- Analytical ε bound may not be tight

Basic empirical privacy auditing [Jagielski et al. 2020]

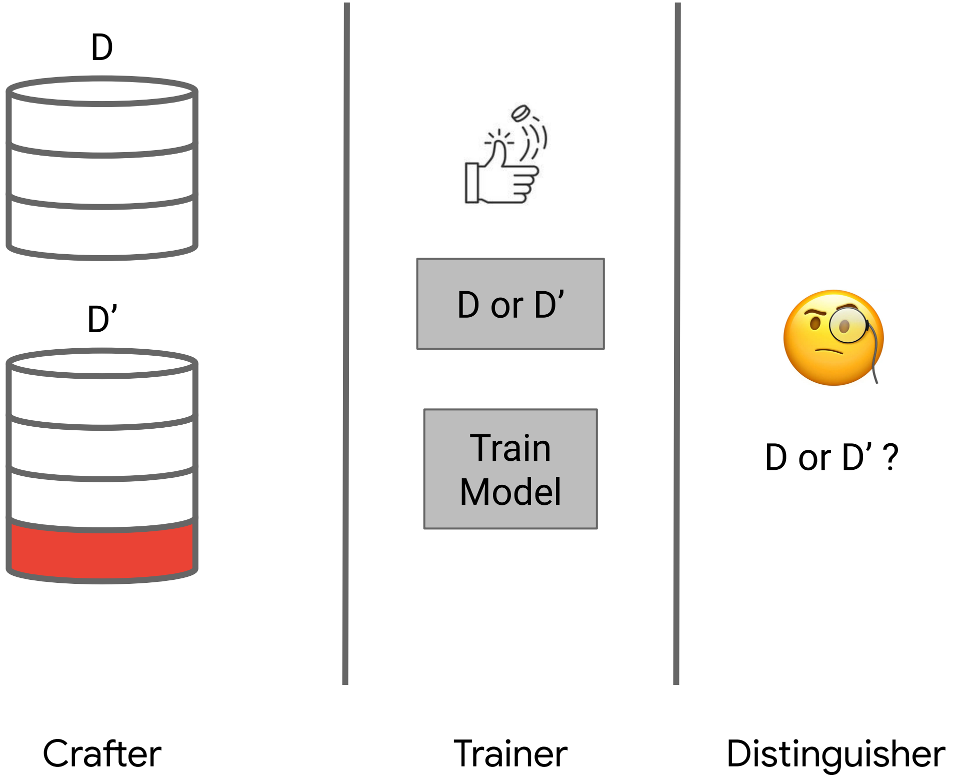


Basic empirical privacy auditing [Jagielski et al. 2020]



Google

Basic empirical privacy auditing [Jagielski et al. 2020]

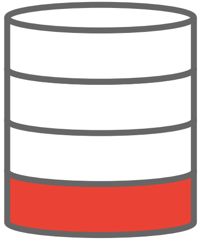


Google

Basic empirical privacy auditing [Jagielski et al. 2020]

D

D’



Crafter

D or D’

Train Model

Trainer

D or D’ ?

Distinguisher

Repeat O(1000) times

Compute optimal

- false positive rate (FPR)
- false negative rate (FNR)

$$\varepsilon = \max\left(\log\frac{1-\delta-FPR}{FNR}, \log\frac{1-\delta-FNR}{FPR}\right)$$

ε lower bound

Basic empirical privacy auditing [Jagielski et al. 2020]

D

D’

Crafter

D or D’

Train Model

Trainer

D or D’ ?

Distinguisher

Repeat
O(1000)
times

Compute optimal

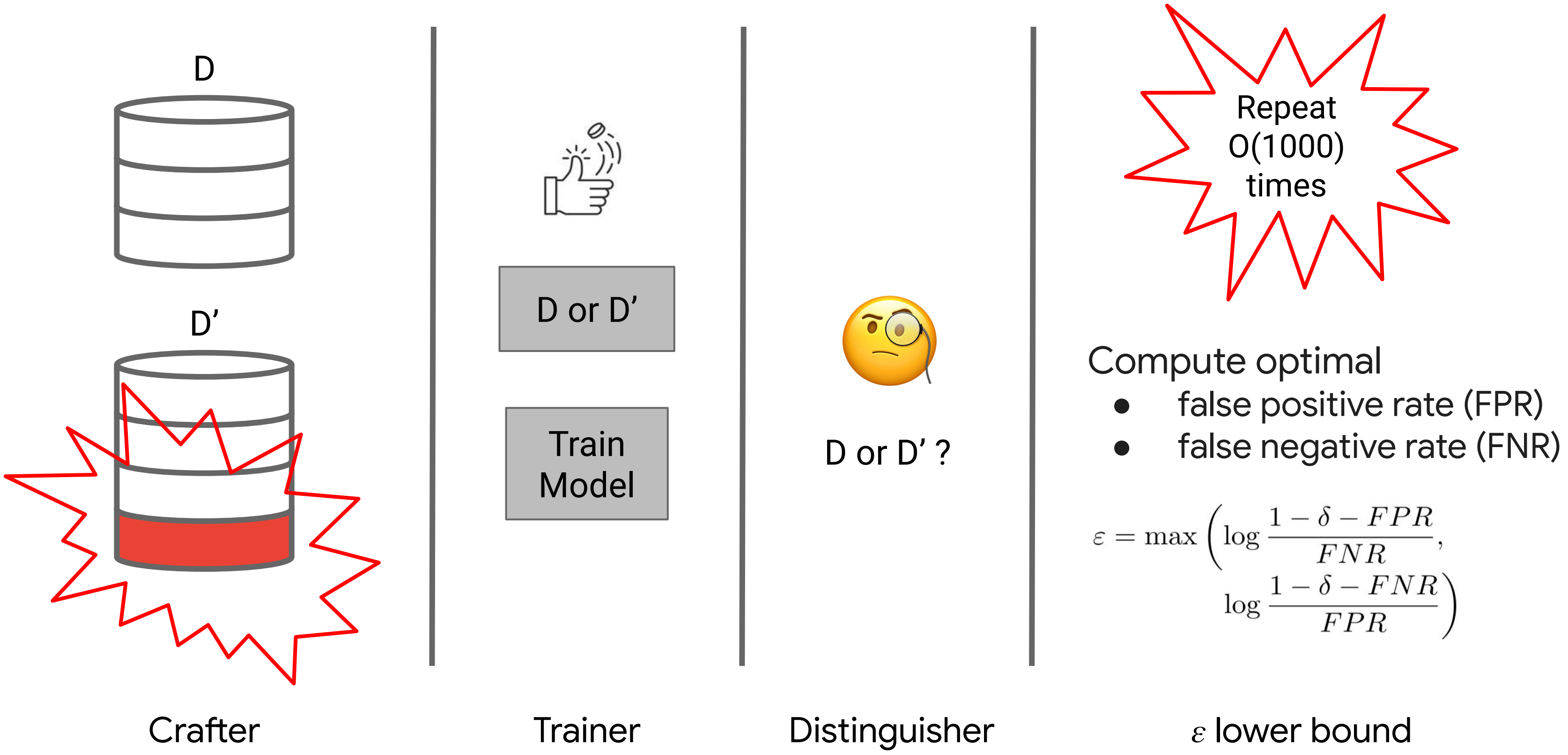
- false positive rate (FPR)
- false negative rate (FNR)

$$\varepsilon = \max\left(\log \frac{1-\delta-FPR}{FNR}, \log \frac{1-\delta-FNR}{FPR}\right)$$

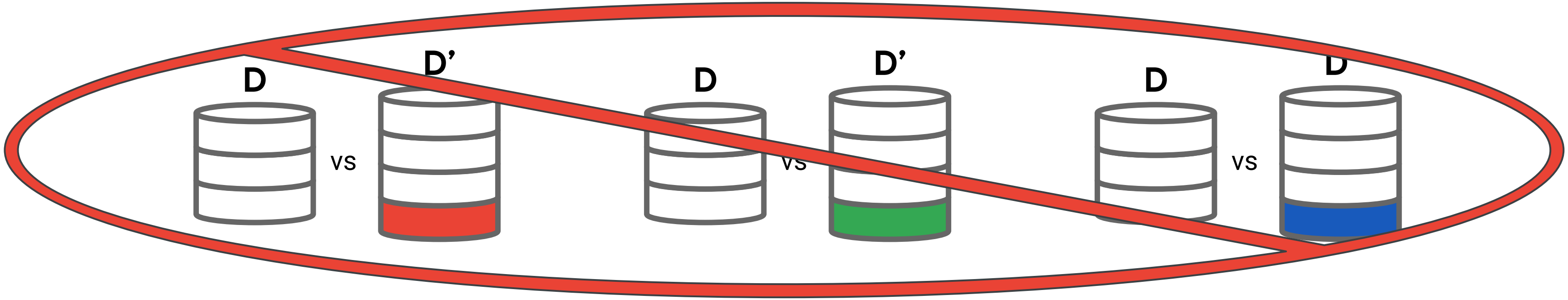
ε lower bound

Google

Basic empirical privacy auditing [Jagielski et al. 2020]

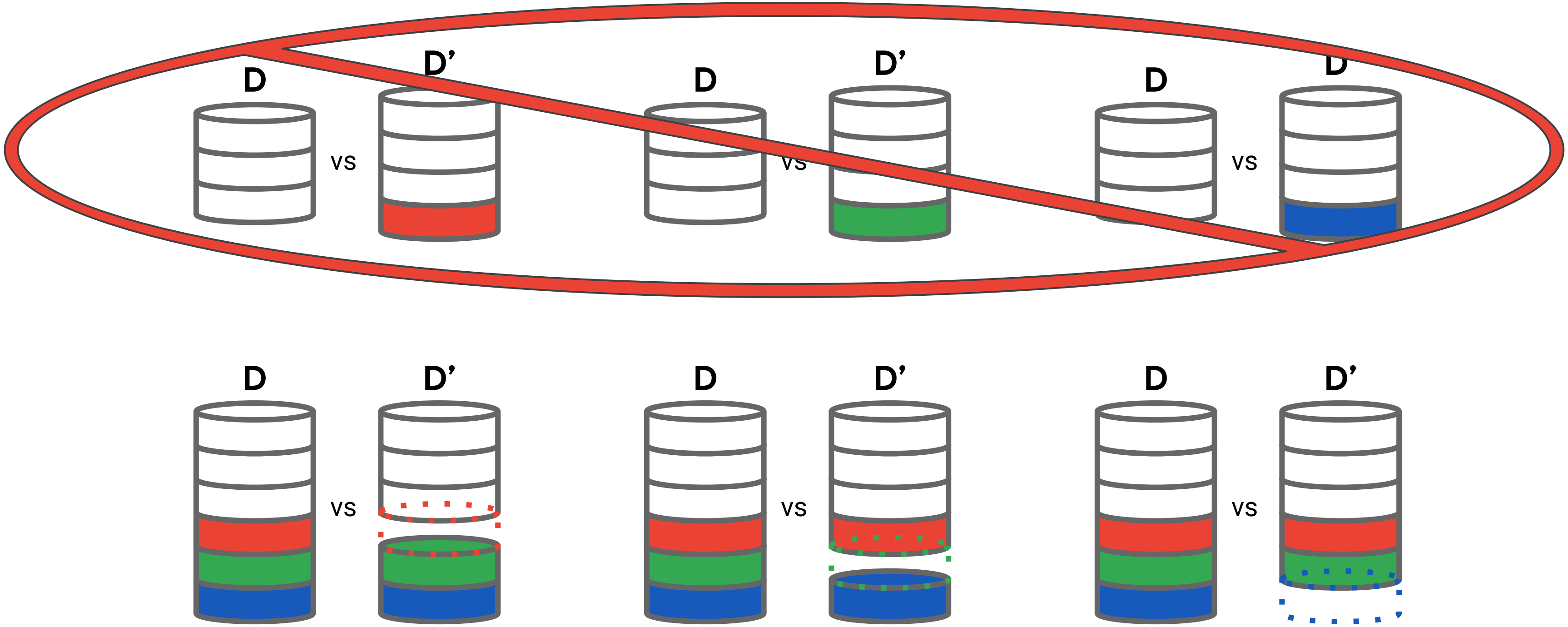


Idea 1: rather than the classical "one canary" in D'



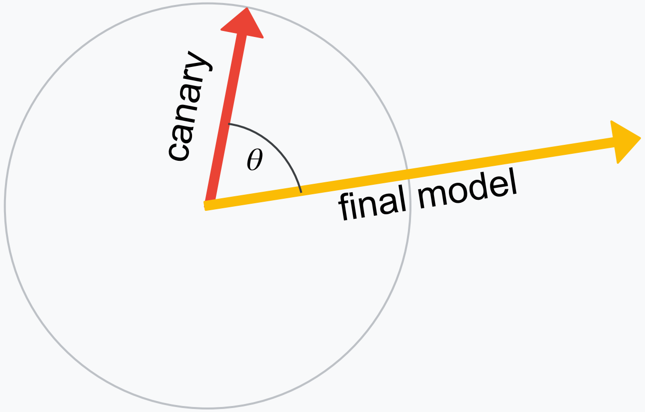
Andrew et al., 2023. One-shot empirical privacy estimation for Federated Learning. ICLR (Oral) 2024

Idea 1: leave one out (LOO) construction of datasets



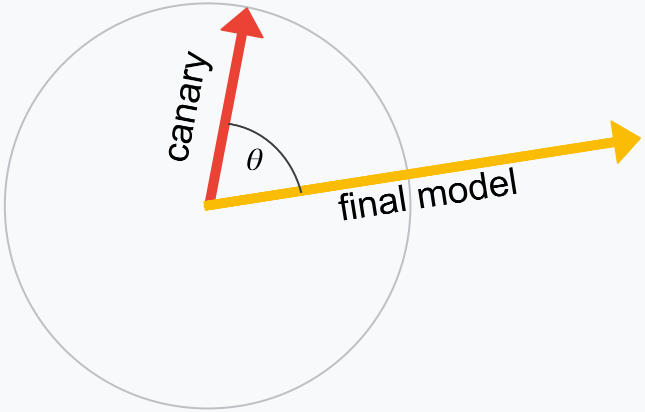
Andrew et al., 2023. One-shot empirical privacy estimation for Federated Learning. ICLR (Oral) 2024

Idea 2: random gradients (user model update) canaries



- Canary updates chosen uniformly from unit d-sphere (model dim d)

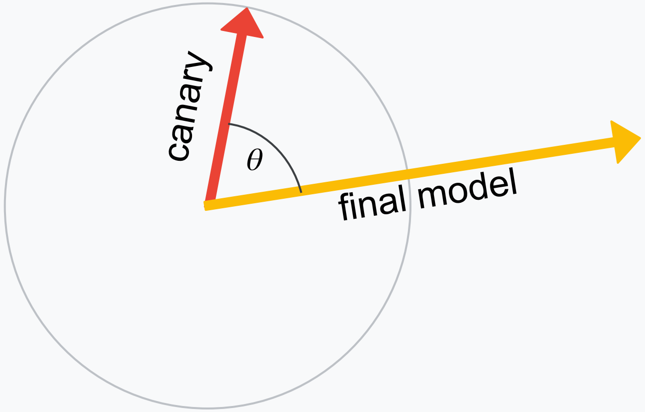
Idea 2: random gradients (user model update) canaries



- Canary updates chosen uniformly from unit d-sphere (model dim d)
- Distinguisher decides based on cosine to final model

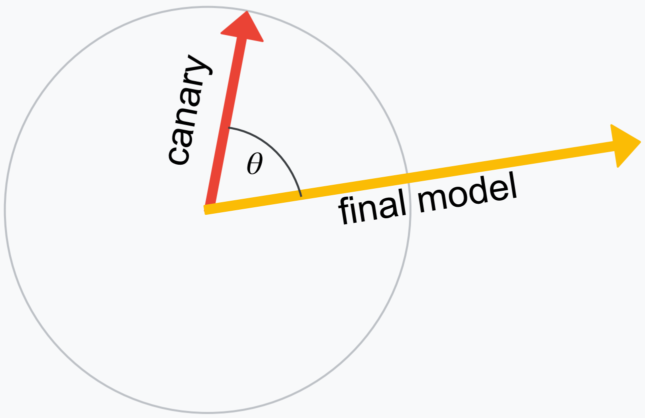
Google

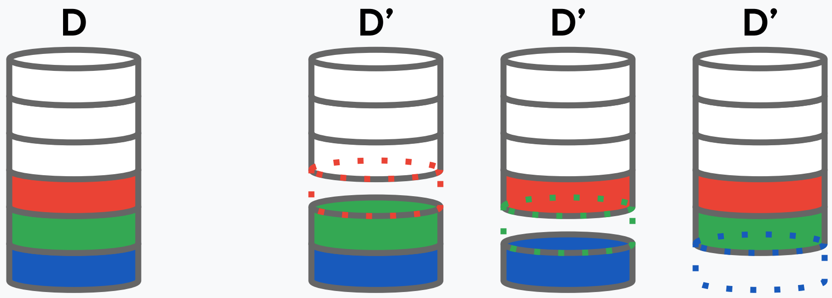
Idea 2: random gradients (user model update) canaries



- Canary updates chosen uniformly from unit d-sphere (model dim d)
- Distinguisher decides based on cosine to final model
- Model memorizes random updates ⇒ higher canary/model cosines ⇒ higher ε estimates

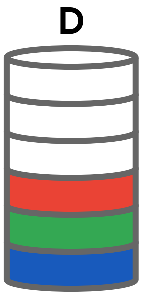
Idea 2: random gradients (user model update) canaries





- Canary updates chosen uniformly from unit d-sphere (model dim d)
- Distinguisher decides based on cosine to final model
- Model memorizes random updates ⇒ higher canary/model cosines ⇒ higher ε estimates
- Null distribution of unobserved canary cosine:
  - does not depend on model
  - *is computable in closed form*

Train on a single dataset with all canaries



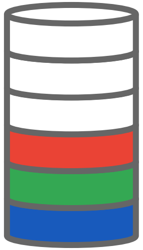
One dataset
with *k* random
canaries
inserted

Crafter

Google

Train on a single dataset with all canaries

D



One dataset with *k* random canaries inserted

Train single model once

Canaries participate in same pattern as real users

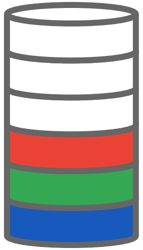
Crafter

Trainer

Google

Train on a single dataset with all canaries

D

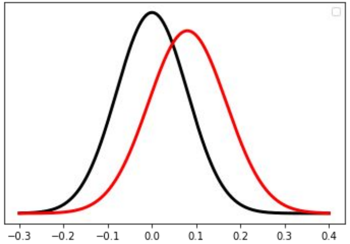


One dataset with *k* random canaries inserted

Train single model once

Canaries participate in same pattern as real users

Null hypothesis cosine distribution $N(0, 1/d)$



Normal approximation to canary cosine sample distribution

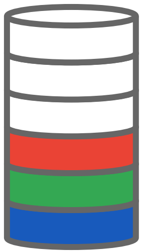
Crafter

Trainer

Distinguisher

Train on a single dataset with all canaries

D



One dataset with k random canaries inserted

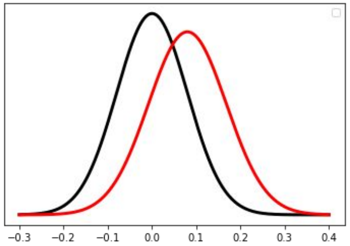
Crafter

Train single model once

Canaries participate in same pattern as real users

Trainer

Null hypothesis cosine distribution $N(0, 1/d)$



Normal approximation to canary cosine sample distribution

Distinguisher

Analytically compute ε comparing two Gaussian distributions

Estimate ε

Google

One-shot method is "correct" for Gaussian Mechanism

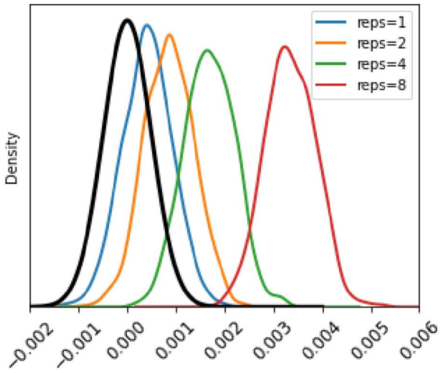
- Gaussian sum is building block of DP-SGD and DP-FedAvg
- Theorem:
  - If model dim d and # of canaries k are high enough (say $d=10^6$, $k=10^3$)
  - Run Gaussian sum mechanism with added canaries
  - Estimate ε of Gaussian mechanism from canary cosine distribution
  - With high probability, recover ε close to the true ε of the mechanism

Experiments on StackOverflow dataset

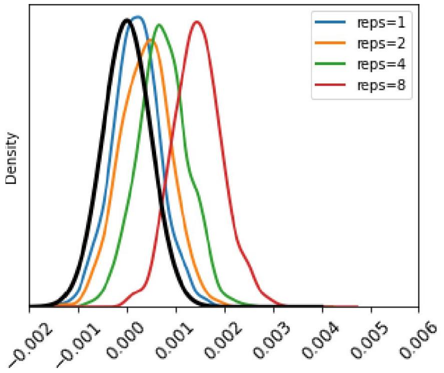
- Model dim 4.1M, 341k clients, one “epoch”
- Replicate canaries 1, 2, 4, 8 times
- Also compare to modified algorithm to estimate ε from all model iterates

Experiments on StackOverflow dataset

- Model dim 4.1M, 341k clients, one "epoch"
- Replicate canaries 1, 2, 4, 8 times
- Also compare to modified algorithm to estimate ε from all model iterates



cosine distribution
clipping only

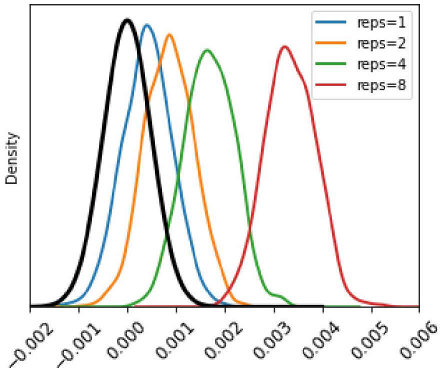


cosine distribution
low DP noise (z=0.050)

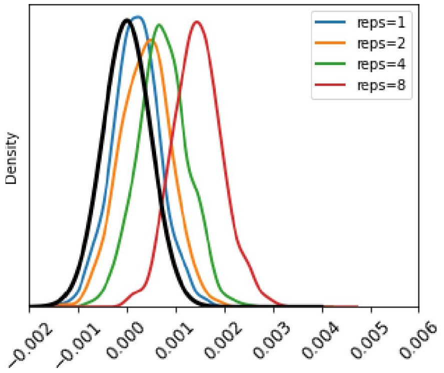
Google

Experiments on StackOverflow dataset

- Model dim 4.1M, 341k clients, one “epoch”
- Replicate canaries 1, 2, 4, 8 times
- Also compare to modified algorithm to estimate ε from all model iterates



cosine distribution
clipping only



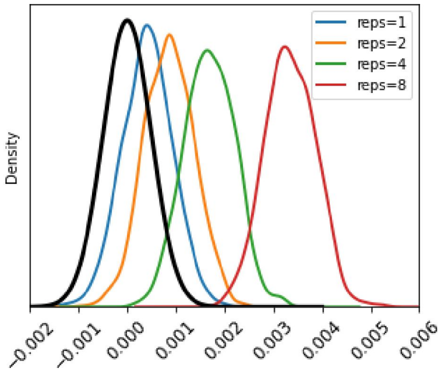
cosine distribution
low DP noise (z=0.050)

Noise	analytical ε	ε_{est}-all	ε_{est}-final
0	∞	45800	4.60
0.050	300	382	1.97
0.099	100	89.4	1.18
0.232	30	2.693	0.569

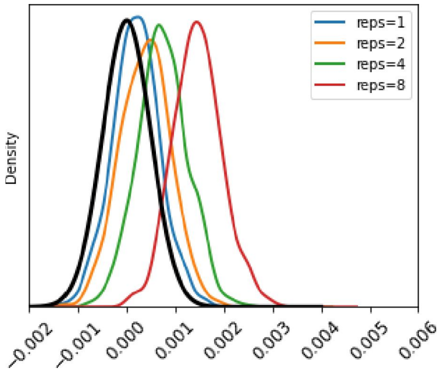
ε estimates, from single repetition of canary.
ε-all uses all model iterates, ε-final uses only final

Experiments on StackOverflow dataset

- Model dim 4.1M, 341k clients, one "epoch"
- Replicate canaries 1, 2, 4, 8 times
- Also compare to modified algorithm to estimate ε from all model iterates



cosine distribution
clipping only



cosine distribution
low DP noise (z=0.050)

Noise	analytical ε	ε_{est}-all	ε_{est}-final
0	∞	45800	4.60
0.050	300	382	1.97
0.099	100	89.4	1.18
0.232	30	2.693	0.569

ε estimates, from single repetition of canary.
ε-all uses all model iterates, ε-final uses only final

Google

Please don’t overfit to DP’s threat model!

Realistic attack, harder for adversary → Less realistic attack, easier for adversary

To ‘win’, adversary must learn …	Many bits of example		One bit of example	One bit about user
During training, adversary controls …	(nothing)	Some examples	Some example gradients	Full dataset
Adversary has access to …	Final model (black box)	Final model (loss)	Final model parameters	All model iterates
Adversary starts with knowledge of …	Short prefix	Complete examples	Full dataset	
Adversary tries to learn about data that is …	Distributed naturally		Out-of-distribution	Any / worst-case
Adversary tries to learn a single secret replicated in …	A single example	All of one user’s examples		All examples of small group

Thank you! Questions?

Krishna Pillutla
Google

Alina Oprea
Northeastern & Google

Galen Andrew
Google

Sewoong Oh
UW & Google

Nikhil Kandpal
U. Toronto

Christopher Choquette
Google

Zheng Xu
Google

Brendan McMahan
Google