Data and Network Security

(Master Degree in Computer Science and Cybersecurity)

Lecture 3

SAPIENZA
UNIVERSITÀ DI ROMA

Outline for today

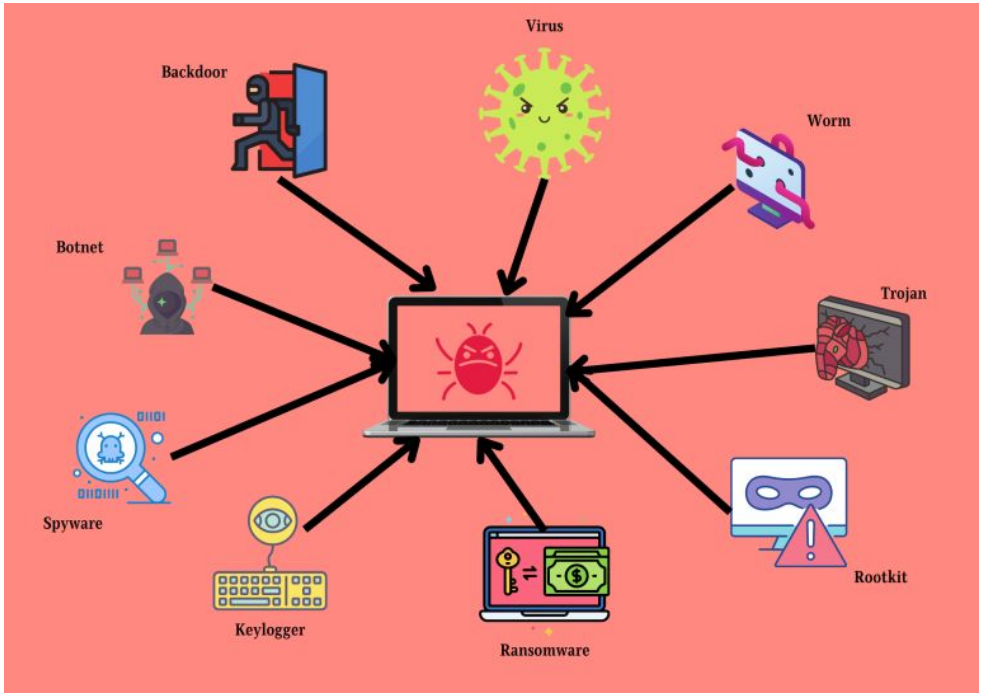
- Recap last lecture
- Covert channels
- Emerging threats

Outline for today

- **Recap last lecture**
- Covert channels
- Emerging threats

Malware

Malicious software designed to infiltrate, damage, or disrupt computer systems, networks, or devices, often with harmful intent.



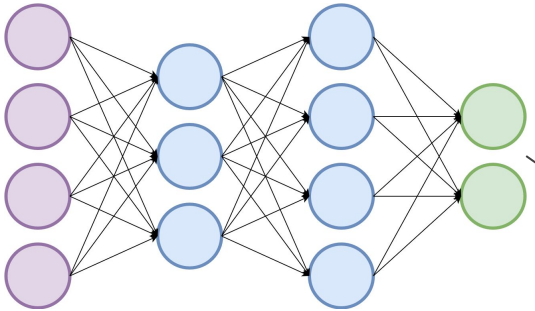
Malware Threats - knowing them

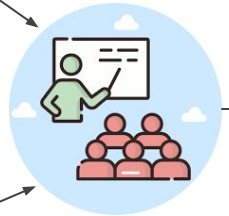
1. Identification and Detection
2. Prevention and mitigation
3. Remediation
4. Risk (assessment and management)
5. General user education
6. Adapting towards evolving threats

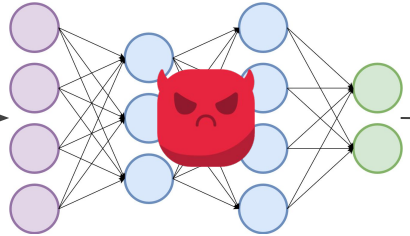
Malware Types - Common

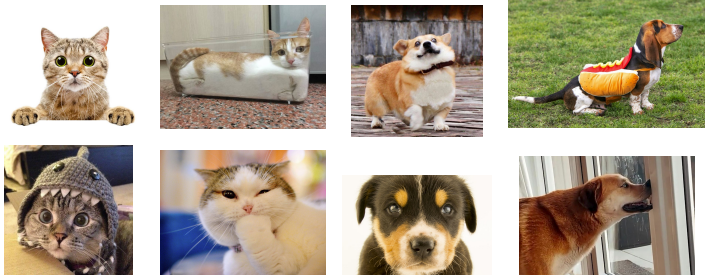
- Viruses
- Worms
- Trojans
- Ransomware
- Spyware
- Adware
- Rootkits
- Scareware

Hiding malware within Deep Neural Networks



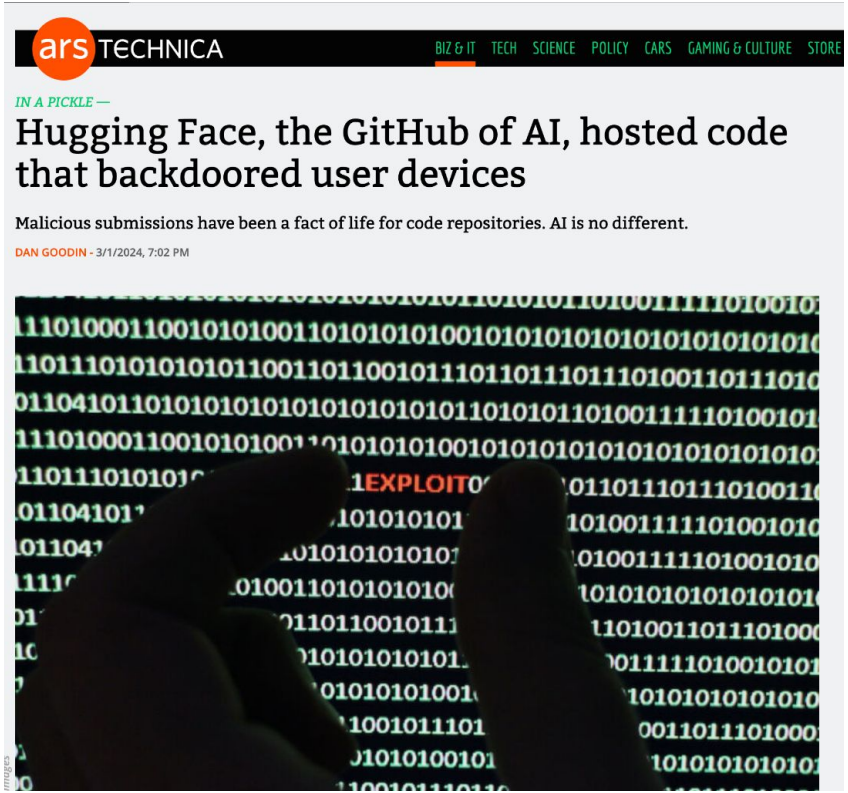






Fresh out of the oven (bad) news...

Hugging Face



ars TECHNICA

BIZ & IT TECH SCIENCE POLICY CARS GAMING & CULTURE STORE

IN A PICKLE —

Hugging Face, the GitHub of AI, hosted code that backdoored user devices

Malicious submissions have been a fact of life for code repositories. AI is no different.

DAN GOODIN - 3/1/2024, 7:02 PM

https://arstechnica.com/security/2024/03/hugging-face-the-github-of-ai-hosted-code-that-backdoored-user-devices/

Outline for today

- Recap last lecture
- **Covert channels**
- Emerging threats

Covert Channel

Indirect communication channel between unauthorized parties that violates some security policy by using **shared resources** in a way in which these resources are not initially designed, bypassing mechanisms that do not permit direct communication between these unauthorized parties in the process.

As such, covert channels emerge as a threat to information-sensitive systems in which leakage to unauthorized parties may be unacceptable.

Covert channel types

- Storage
- Timing

Storage based covert channel

Covert channels that exploit storage resources to conceal data, often utilizing file attributes or reserved storage space.

Storage based covert channel

Covert channels that exploit storage resources to conceal data, often utilizing file attributes or reserved storage space.

- Data hidden within file (such as steganography)
- Modifying header fields

Storage based covert channel - Detection & Mitigation

- Monitoring file system activity
- Analyze file attributes
- Integrity checks to identify anomalies or unauthorized data storage
- Access control

Timing based covert channel

Covert channels that exploit variations in timing or delays within a standard communication channel to conceal data.

Modulating inter-packet delays

- Large delay - bit 1
- Small delay - bit 0

Timing based covert channel - Detection and Mitigation

Covert channels that exploit variations in timing or delays within a standard communication channel to conceal data.

- Detecting:
  - Modulated traffic would have different IDP distribution
- Mitigation
  - Injecting random delays
  - Buffering

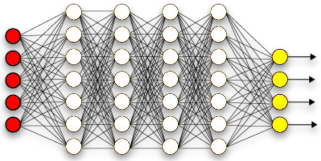
Outline for today

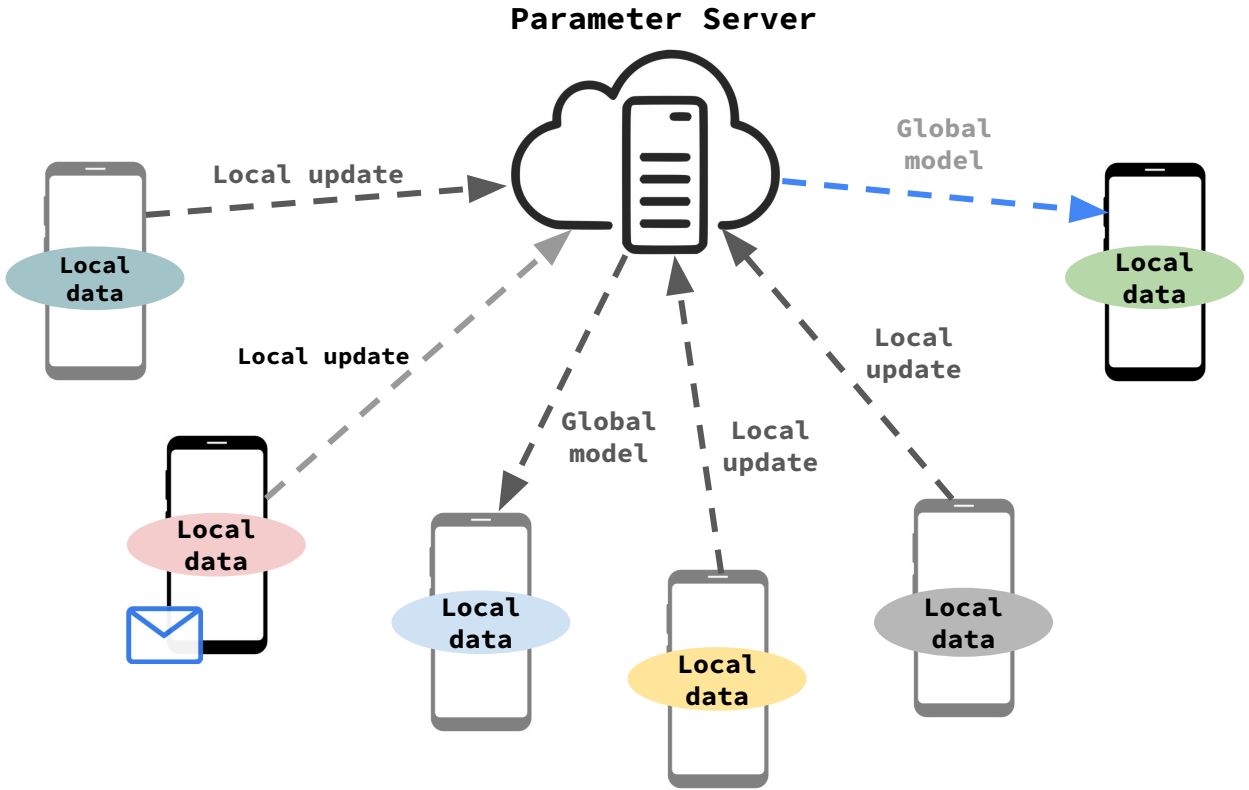
- Recap last lecture
- Covert channels
- **Emerging threats**

Covert Channels in Collaborative (federated) learning

Collaborative Learning

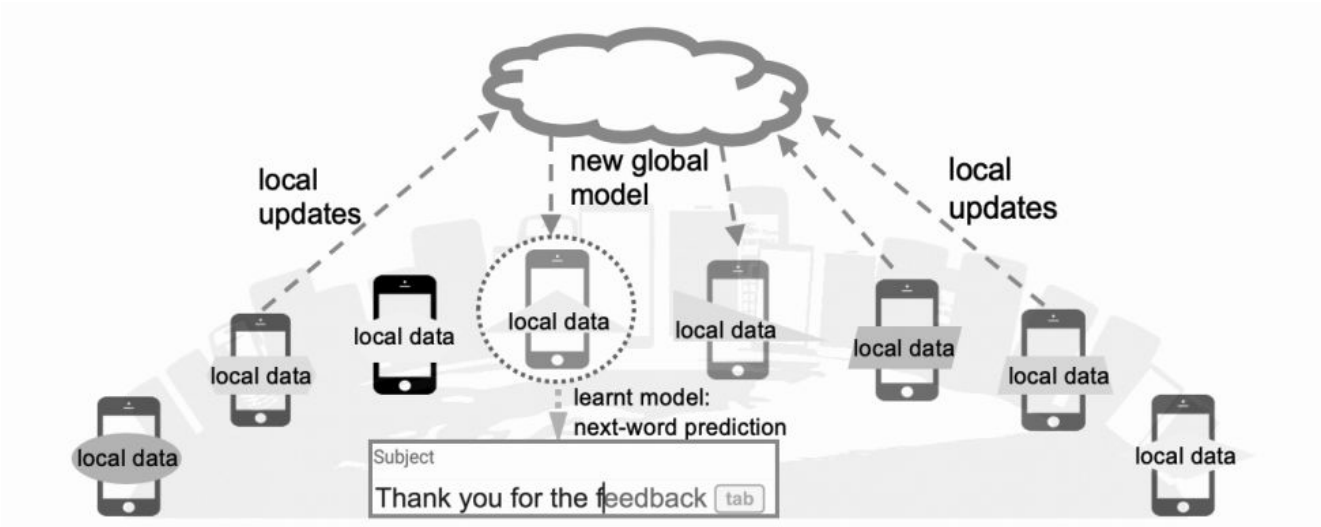
Collaborative Learning





What is Federated Learning?

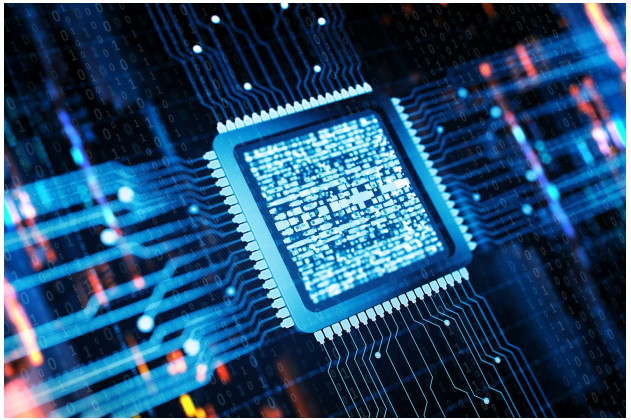
Google AI



Federated learning (FL) (also known as **collaborative learning**) is a machine learning technique that trains an algorithm across multiple decentralized edge devices or servers holding local data samples, without exchanging them.

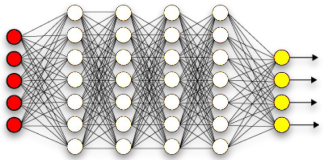
Why federated Learning?

- Data Privacy
- Low individual computing power
- Large collaborative computing power

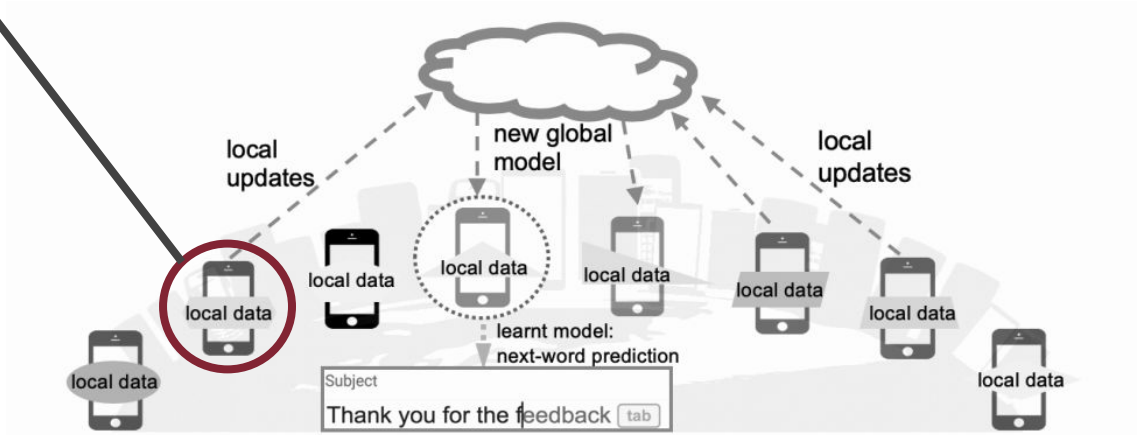


How does FL work? (local update)

$$\mathbf{W}_{t+1}^{k} = \mathbf{W}_t + \alpha \nabla \mathbf{W}_t^{k}$$

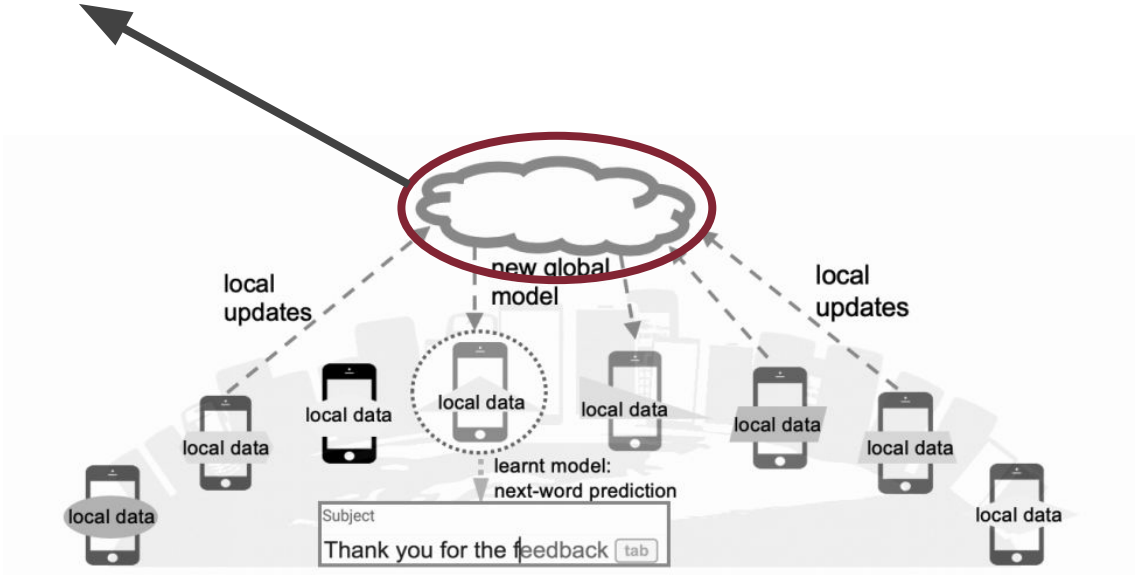


W



How does FL work? (global update)

$$\mathbf{W}_{t+1} = \mathbf{W}_t + \frac{\alpha}{n'} \sum_{k=1}^{n'} \nabla \mathbf{W}_t^k$$

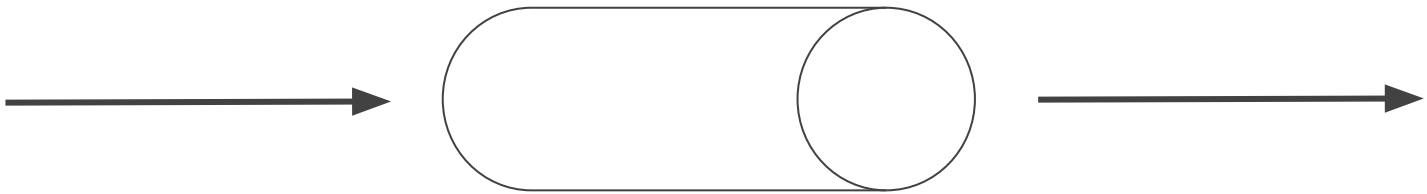


Digital (stealthy) communication

- Recap from lecture 2-

Digital Communication

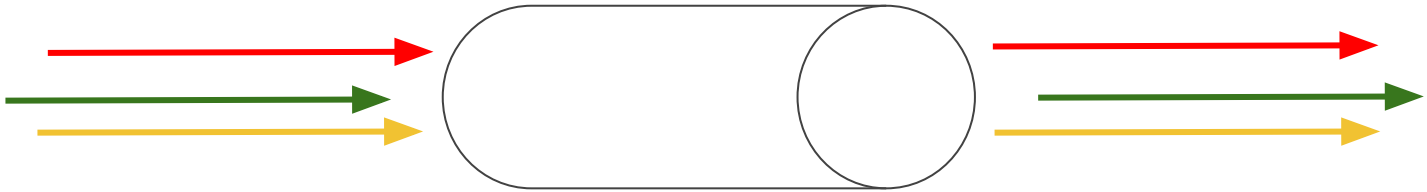
- Consists of the transfer of data or information using digital signals over a point-to-point channel.
- Data or information are digitally encoded as discrete signals, and are transferred through a communication channel.



Communication channel

What is CDMA?

- Channel coding technique by which a narrowband signal is spread into a wider bandwidth.
- Codes the data at a higher frequency by using pseudorandomly generated codes that the receiver knows.
- developed in the 1950s for military communications:
  - resist the enemy efforts to jam the communication channel.
  - hide the fact that the communication is taking place.



Communication channel

CDMA - example

Binary sequence: **[0, 1, 1]**

PSK **[-1, 1, 1]**

Spreading code: **[-1, 1, -1, -1, 1]**

Chip sequence: **[1, -1, 1, 1, -1, -1, 1, -1, -1, 1, -1, 1, -1, -1, 1]**

-5 **+5** **+5**

let's send some “message”

HowTo

Payload -> **P** bits **b** = $[b_0, \ldots, b_{P-1}]$

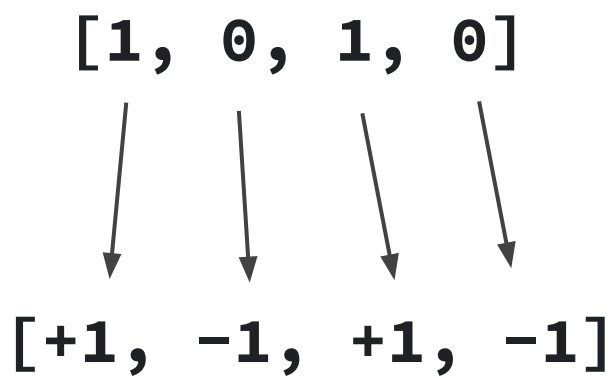
- Each bit is encoded as **±1**.

$$[1, 0, 1, 0]$$

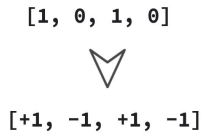
HowTo (cont.)

Payload -> **P** bits **b** = $[b_0, \ldots, b_{P-1}]$

- Each bit is encoded as **±1**.



HowTo (cont.)



Payload -> **P** bits **b** = [b_0, …,b_{P-1}]

- The spreading code of each bit (c_i) is a vector of ±1 that is of the same length of vector W, namely R.

[+1, −1, +1, −1]

[+1, +1, −1, +1, …,−1] c_i

R elements

HowTo (cont.)

Payload -> P bits b = [b_0, …,b_{P-1}]

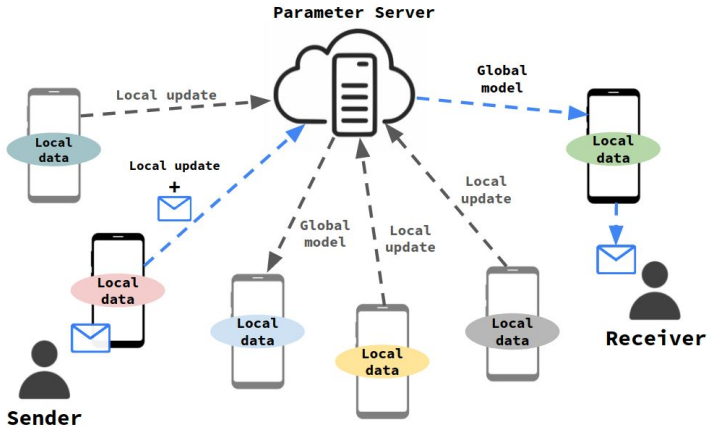
- **C** is an **R** by **P** matrix that collects all the codes.

	+1	**+1**	**-1**		**+1**
	-1	**-1**	**-1**		**-1**
	+1	**-1**	**+1**		**+1**
R	**.**	**.**	**.**		**.**
	.	**.**	**.**	**. . .**	**.**
	.	**.**	**.**		**.**
	-1	**-1**	**-1**		**+1**

P

“message” embedding

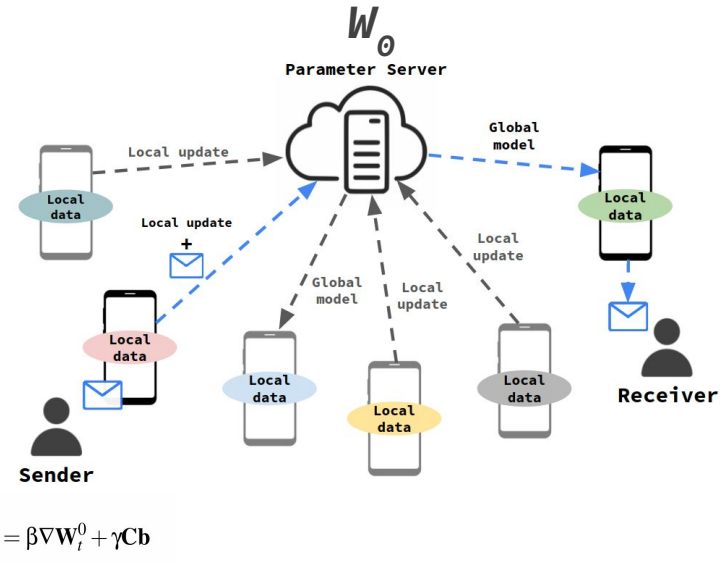
- **n** participants
- Parameter server proposes a set of weights W_0
- At each iteration, the participants use their local data to compute the gradient ∇W_t^k k = 0, …, n-1; t = 0, . . . T - 1



“message” embedding (cont.)

$$\widehat{\nabla \mathbf{W}_t^0} = \beta \nabla \mathbf{W}_t^0 + \gamma \mathbf{C}\mathbf{b}$$

The gradient update of the sender



γ and **β** are two gain factors to ensure that the message cannot be detected and that the power of the modified gradient is like the unmodified gradient for the other users.

“message” embedding (cont.)

How do we choose **γ** and **β** ?

$$\widehat{\nabla \mathbf{W}_t^0} = \beta \nabla \mathbf{W}_t^0 + \gamma \mathbf{C}\mathbf{b}$$

“message” embedding (cont.)

How do we choose **γ** and **β** ?

$$\widehat{\nabla \mathbf{W}_t^0} = \beta \nabla \mathbf{W}_t^0 + \gamma \mathbf{C}\mathbf{b}$$

β=0 and **γ=σ/√P**

- Our gradient would have the same power as the original.
- A hypothesis testing looking for a binomial or a Gaussian distribution will be able to detect that our gradient is not a true gradient.

“message” embedding (cont.)

How do we choose **γ** and **β** ?

$$\widehat{\nabla \mathbf{W}_t^0} = \beta \nabla \mathbf{W}_t^0 + \gamma \mathbf{C}\mathbf{b}$$

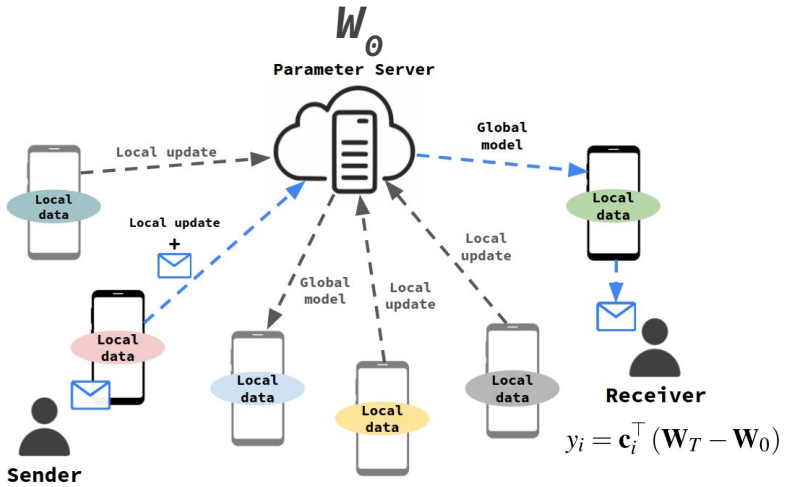
β=1 and **γ=0.1σ/√P**

- Our gradient will have the same distribution as the original gradient.
- The signal will be undetectable.

“message” extraction

To recover bit i of the payload:

$$y_i = \mathbf{c}_i^\top (\mathbf{W}_T - \mathbf{W}_0)$$



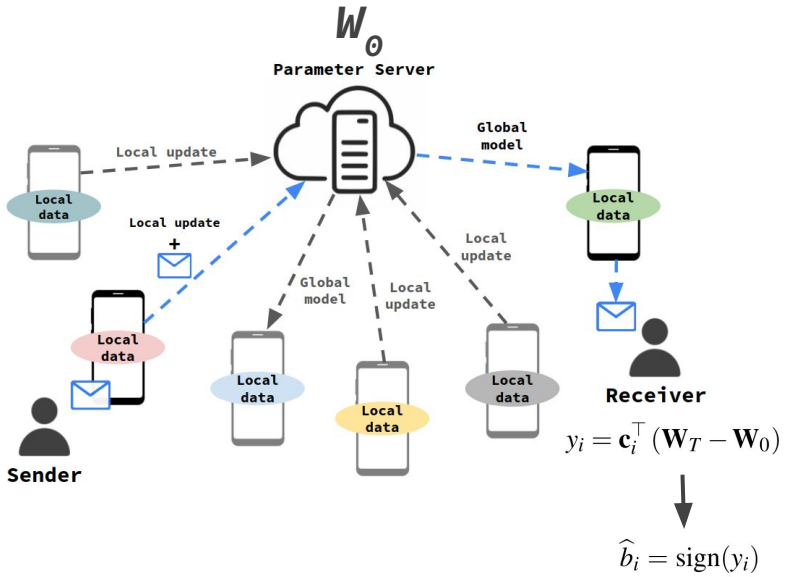
1. Bit i signal.
2. Noise from the gradients.
3. Noise from the other bits of the payload.

“message” extraction

To recover bit i of the payload:

$$y_i = \mathbf{c}_i^\top (\mathbf{W}_T - \mathbf{W}_0)$$

$$\widehat{b}_i = \text{sign}(y_i)$$



How we recover b_i ?

How we recover b_i ?

$$y_i = \mathbf{c}_i^\top (\mathbf{W}_T - \mathbf{W}_0)$$

Assume $\nabla \mathbf{W}_t^k$ is a zero-mean with a variance σ^2

How we recover b_i ?

$$
\begin{aligned}
y_i =& \mathbf{c}_i^\top \left(\mathbf{W}_T - \mathbf{W}_0\right) \\
=& \frac{\alpha}{n} \mathbf{c}_i^\top \left(\sum_{t=0}^{T-1} \left(\widehat{\nabla \mathbf{W}_t^0} + \sum_{k=1}^{n-1} \nabla \mathbf{W}_t^k \right) \right) \\
=& \frac{\alpha}{n} \left(T\gamma \mathbf{c}_i^\top \mathbf{C}\mathbf{b} + \mathbf{c}_i^\top \sum_{t=0}^{T-1} \left(\beta \nabla \mathbf{W}_0^k + \sum_{k=0}^{n-1} \nabla \mathbf{W}_t^k \right) \right) \\
=& \frac{\alpha}{n} \left(T\gamma \mathbf{c}_i^\top \mathbf{c}_i b_i + T\gamma \mathbf{c}_i^\top \mathbf{C}_{\neg i} \mathbf{b}_{\neg i} + \mathbf{c}_i^\top \widetilde{\mathbf{w}} \right) \\
=& \frac{\alpha}{n} \left(T\gamma \mathbf{c}_i^\top \mathbf{c}_i b_i + T\gamma \mathbf{c}_i^\top \widetilde{\mathbf{c}} + \mathbf{c}_i^\top \widetilde{\mathbf{w}} \right) \\
=& \frac{\alpha}{n} \left(T\gamma R b + \boldsymbol{\varepsilon}_i^{\mathbf{C}} + \boldsymbol{\varepsilon}_i^{\mathbf{W}} \right) = \frac{\alpha}{n} \left(T\gamma R b + \boldsymbol{\varepsilon}_i \right)
\end{aligned}
$$

How we recover b_i ?

$$
\begin{aligned}
y_i =& \mathbf{c}_i^\top \left(\mathbf{W}_T - \mathbf{W}_0\right) \\
=& \frac{\alpha}{n} \mathbf{c}_i^\top \left(\sum_{t=0}^{T-1} \left(\widehat{\nabla \mathbf{W}_t^0} + \sum_{k=1}^{n-1} \nabla \mathbf{W}_t^k \right) \right) \\
=& \frac{\alpha}{n} \left(T\gamma \mathbf{c}_i^\top \mathbf{C}\mathbf{b} + \mathbf{c}_i^\top \sum_{t=0}^{T-1} \left(\beta \nabla \mathbf{W}_0^k + \sum_{k=0}^{n-1} \nabla \mathbf{W}_t^k \right) \right) \\
=& \frac{\alpha}{n} \left(T\gamma \mathbf{c}_i^\top \mathbf{c}_i b_i + T\gamma \mathbf{c}_i^\top \mathbf{C}_{\neg i} \mathbf{b}_{\neg i} + \mathbf{c}_i^\top \widetilde{\mathbf{w}} \right) \\
=& \frac{\alpha}{n} \left(T\gamma \mathbf{c}_i^\top \mathbf{c}_i b_i + T\gamma \mathbf{c}_i^\top \widetilde{\mathbf{c}} + \mathbf{c}_i^\top \widetilde{\mathbf{w}} \right) \\
=& \frac{\alpha}{n} \left(T\gamma R b + \varepsilon_i^{\mathbf{C}} + \varepsilon_i^{\mathbf{W}} \right) = \frac{\alpha}{n} \left(T\gamma R b + \varepsilon_i \right)
\end{aligned}
$$

- ~c is a symmetric binomial distribution between **±(P - 1).**
- Binomial distribution with values between **±R(P-1).**
- For large **R** can be approximated by a zero-mean Gaussian with variance $\mathbf{T^2\gamma^2R(P-1)}$

How we recover b_i ?

$$
\begin{aligned}
y_i &= \mathbf{c}_i^\top (\mathbf{W}_T - \mathbf{W}_0) \\
&= \frac{\alpha}{n} \mathbf{c}_i^\top \left(\sum_{t=0}^{T-1} \left(\widehat{\nabla \mathbf{W}_t^0} + \sum_{k=1}^{n-1} \nabla \mathbf{W}_t^k \right) \right) \\
&= \frac{\alpha}{n} \left(T\gamma \mathbf{c}_i^\top \mathbf{C}\mathbf{b} + \mathbf{c}_i^\top \sum_{t=0}^{T-1} \left(\beta \nabla \mathbf{W}_0^k + \sum_{k=0}^{n-1} \nabla \mathbf{W}_t^k \right) \right) \\
&= \frac{\alpha}{n} \left(T\gamma \mathbf{c}_i^\top \mathbf{c}_i b_i + T\gamma \mathbf{c}_i^\top \mathbf{C}_{\neg i} \mathbf{b}_{\neg i} + \mathbf{c}_i^\top \widetilde{\mathbf{w}} \right) \\
&= \frac{\alpha}{n} \left(T\gamma \mathbf{c}_i^\top \mathbf{c}_i b_i + T\gamma \mathbf{c}_i^\top \widetilde{\mathbf{c}} + \mathbf{c}_i^\top \widetilde{\mathbf{w}} \right) \\
&= \frac{\alpha}{n} \left(T\gamma R b + \varepsilon_i^{\mathbf{C}} + \varepsilon_i^{\mathbf{W}} \right) = \frac{\alpha}{n} \left(T\gamma R b + \varepsilon_i \right)
\end{aligned}
$$

- Each component of $\widetilde{\mathbf{w}}$ adds up **$T(n-1+\beta)$** of these values
- By CLT (large enough **$T(n-1+\beta)$**) each one of these variables would be zero-mean Gaussian with a variance **$Tn\sigma^2$**.
- When we multiply this vector by **$\mathbf{c}_i$** and add all the components together, we end up with a zero-mean Gaussian with a variance **$RTn\sigma^2$**, because the components of **$\mathbf{c}_i$** are ±1.

How we recover b_i?

How we recover b_i?

$$
\begin{aligned}
y_i = \mathbf{c}_i^\top (\mathbf{W}_T - \mathbf{W}_0) &= \frac{\alpha}{n}\left(T\gamma \mathbf{c}_i^\top \mathbf{c}_i b_i + T\gamma \mathbf{c}_i^\top \mathbf{C}_{\neg i}\mathbf{b}_{\neg i} + \mathbf{c}_i^\top \widetilde{\mathbf{w}}\right) \\
&= \frac{\alpha}{n}\left(T\gamma \mathbf{c}_i^\top \mathbf{c}_i b_i + T\gamma \mathbf{c}_i^\top \widetilde{\mathbf{c}} + \mathbf{c}_i^\top \widetilde{\mathbf{w}}\right) \\
&= \frac{\alpha}{n}\left(T\gamma R b + \varepsilon_i^{\mathbf{C}} + \varepsilon_i^{\mathbf{W}}\right) = \frac{\alpha}{n}\left(T\gamma R b + \varepsilon_i\right)
\end{aligned}
$$

- **C**, **b** and $\nabla \mathbf{W}_t^k$ are mutually independent, thus $\boldsymbol{\varepsilon}_i$ is zero mean with a variance that it is the sum of the variances of $\boldsymbol{\varepsilon}^{\mathbf{C}}_i$ and $\boldsymbol{\varepsilon}^{\mathbf{W}}_i$ and also Gaussian distributed.
- The distribution of $\mathbf{y}_i$ is given by $\mathbf{y}_i \sim \mathcal{N}(\mathbf{T\gamma R b}_i, \mathbf{T^2\gamma^2 R(P-1) + RTn\sigma^2})$, and if normalize by $\mathbf{T\gamma R}$:

$$
y_i \sim \mathcal{N}\left(b_i, \frac{T^2\gamma^2 R(P-1) + TRn\sigma^2}{T^2\gamma^2 R^2}\right)
$$

$$
\mathcal{N}\left(b_i, \frac{P-1}{R} + \frac{n\sigma^2}{TR\gamma^2}\right)
$$

How we recover b_i?

$$y_i \sim \mathcal{N}\left(b_i, \frac{T^2\gamma^2 R(P-1) + TRn\sigma^2}{T^2\gamma^2 R^2}\right)$$

$$\mathcal{N}\left(b_i, \frac{P-1}{R} + \frac{n\sigma^2}{TR\gamma^2}\right)$$

Using a long enough error-correcting code, we can ensure errorless communication when the variance is about **1**.

β=1 and **γ=0.1σ/√P**

$$\frac{P-1}{R} + \frac{n\sigma^2}{TR\gamma^2} = \frac{P-1}{R} + \frac{n\sigma^2}{TR\left(\frac{0.1\sigma}{\sqrt{P}}\right)^2}$$

$$= \frac{P-1}{R} + \frac{100nP\sigma^2}{TR} \approx \frac{(T+100n)P}{TR}$$

- At least **T>100nP/(R-P)** rounds before the message can be decoded.

How we recover b_i faster?

- We incorporate multiple senders

How we recover b_i faster?

- We incorporate multiple senders

  - The y_i distribution will become:

$$\mathcal{N}(MT\gamma Rb_i, M^2T^2\gamma^2R(P-1) + RTn\sigma^2)$$

How we recover b_i faster?

- We incorporate multiple senders

  - The y_i distribution will become:

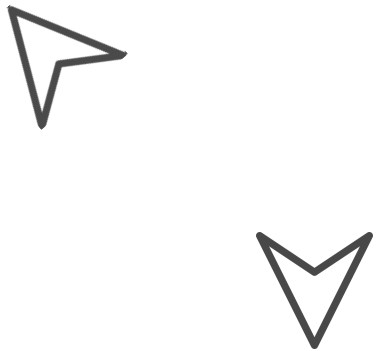
$$\mathcal{N}(MT\gamma Rb_i, M^2T^2\gamma^2R(P-1)+RTn\sigma^2)$$

$T>nP/M^2(R-P)$ M^2 times faster

The pillars of evaluation

Stealthiness of the communication

Impact on model performance

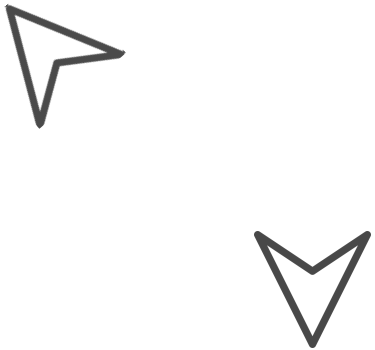


“Message” delivery time

The pillars of evaluation

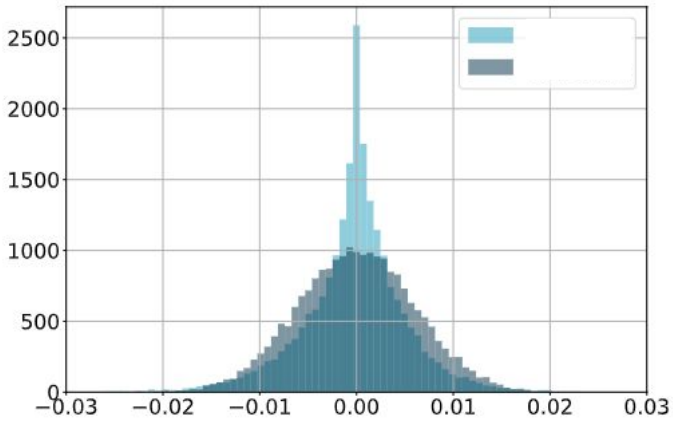
Stealthiness of the communication

Impact on model performance



“Message” delivery time

Stealthiness of communication



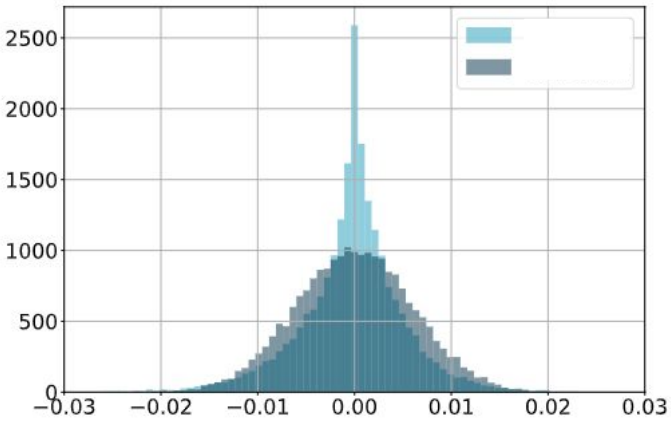
Regular gradient update

vs.

Non-Stealthy gradient update

In **non-stealthy** we are not sending a gradient that is useful for learning, instead we send our signal with the same power as our gradient would have.

Stealthiness of communication



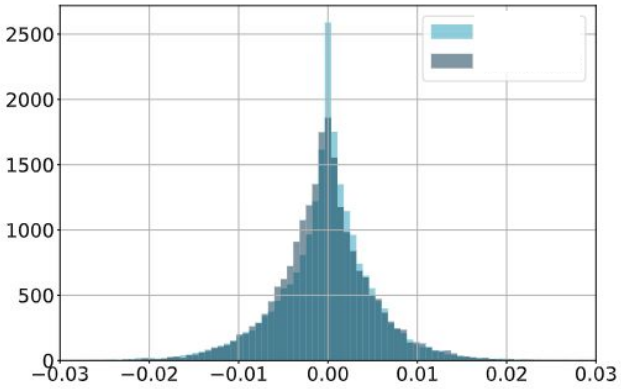
Regular gradient update

vs.

Non-Stealthy gradient update

NON-STEALTHY **might** be detectable in cases where the global parameter server can observe individual gradient updates.

Stealthiness of communication



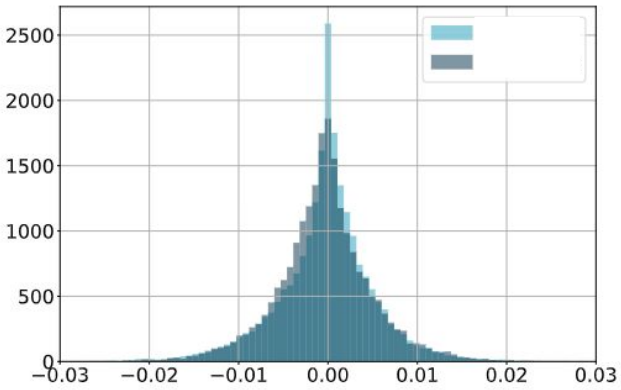
Regular gradient update

vs.

Full-Stealthy gradient update

In **full-stealthy** we are sending a gradient that is useful for learning, and buried into it we put also our signal.

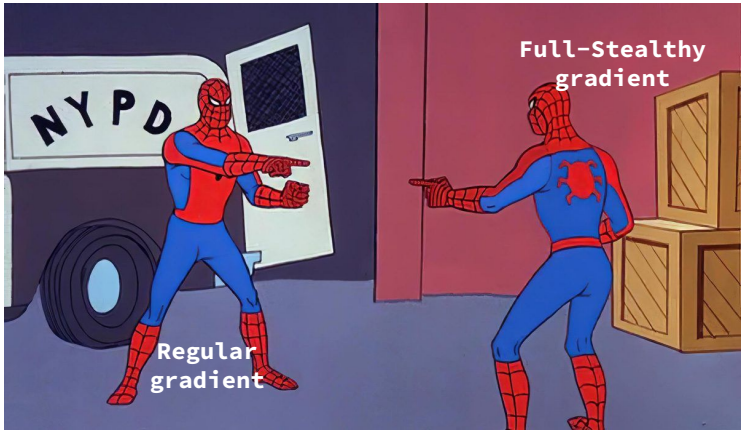
Stealthiness of communication



Regular gradient update

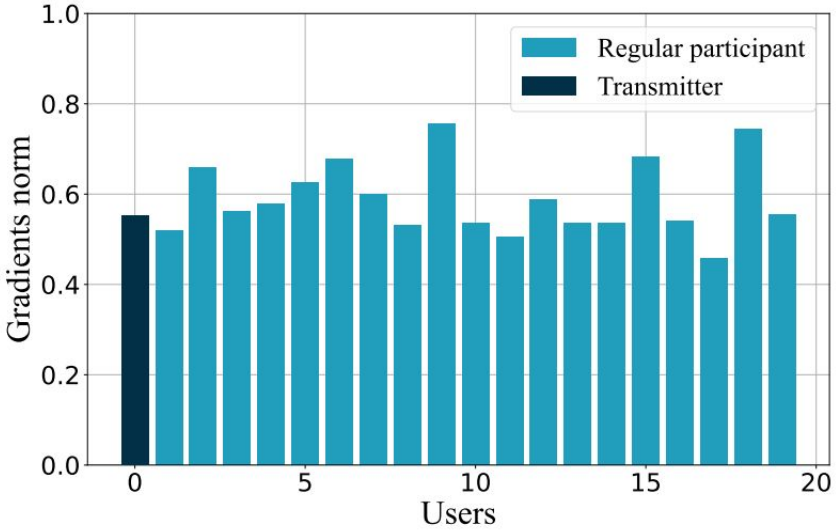
vs.

Full-Stealthy gradient update



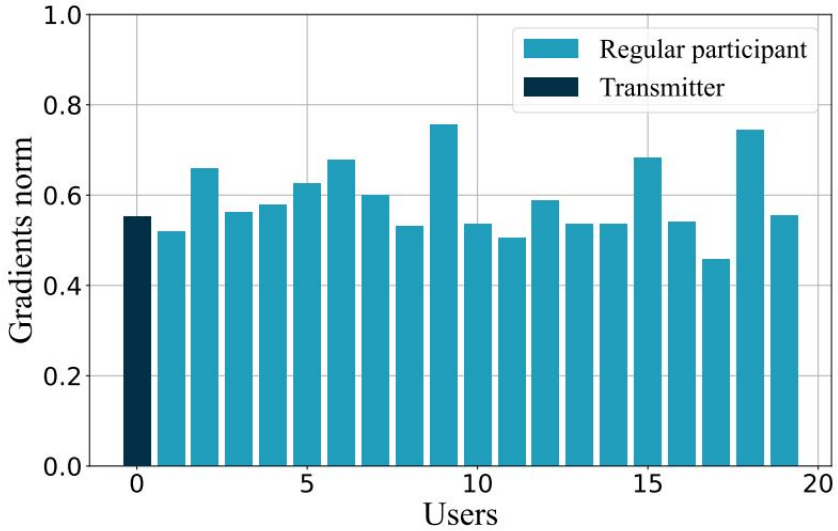
Gradients are statistically indistinguishable

Stealthiness of communication (cont.)



Parameter server observes individual gradient updates

Stealthiness of communication (cont.)

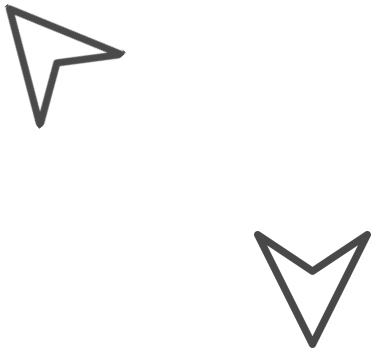


Parameter server observes individual gradient updates

The pillars of evaluation

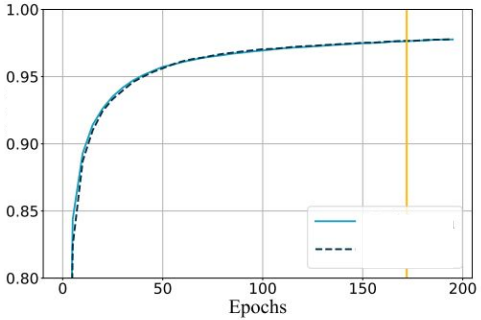
Stealthiness of the communication

Impact on model performance



“Message” delivery time

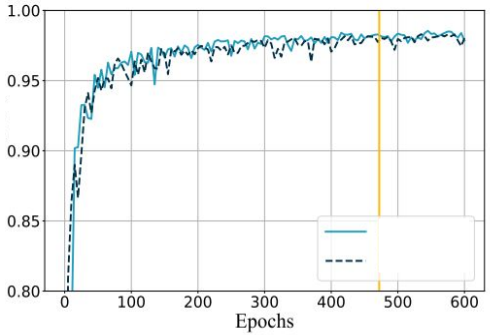
Impact on the model performance



100 participants

10 senders FULL Stealthy

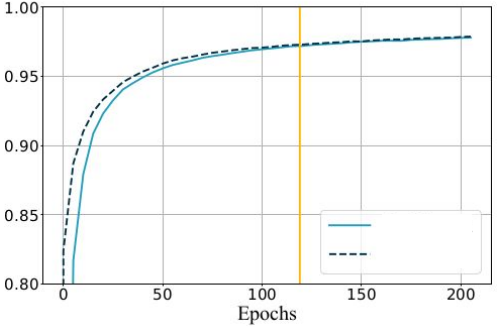
100% aggregated per round



100 participants

10 senders FULL Stealthy

20% aggregated per round



100 participants

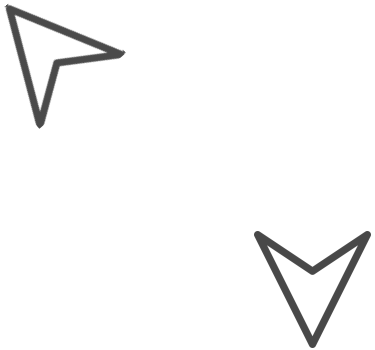
1 sender NON Stealthy

100% aggregated per round

The pillars of evaluation

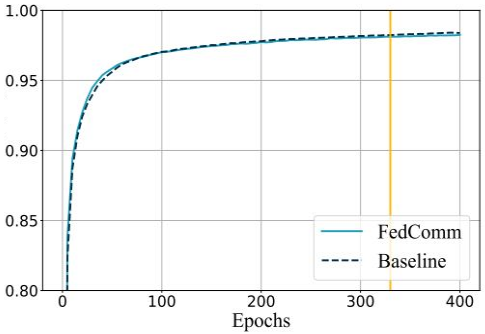
Stealthiness of the communication

Impact on model performance



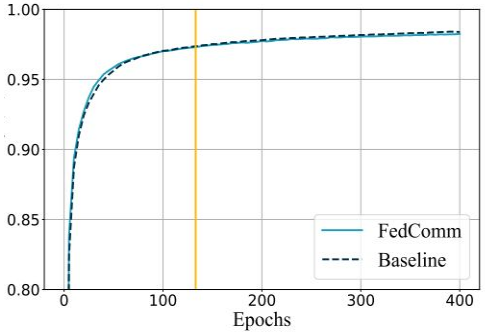
“Message” delivery time

“message” delivery time



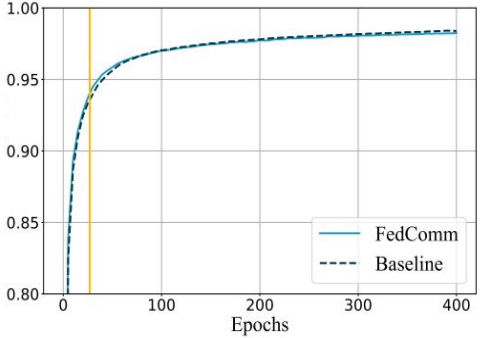
1 sender

100% aggregated per round



2 senders

100% aggregated per round

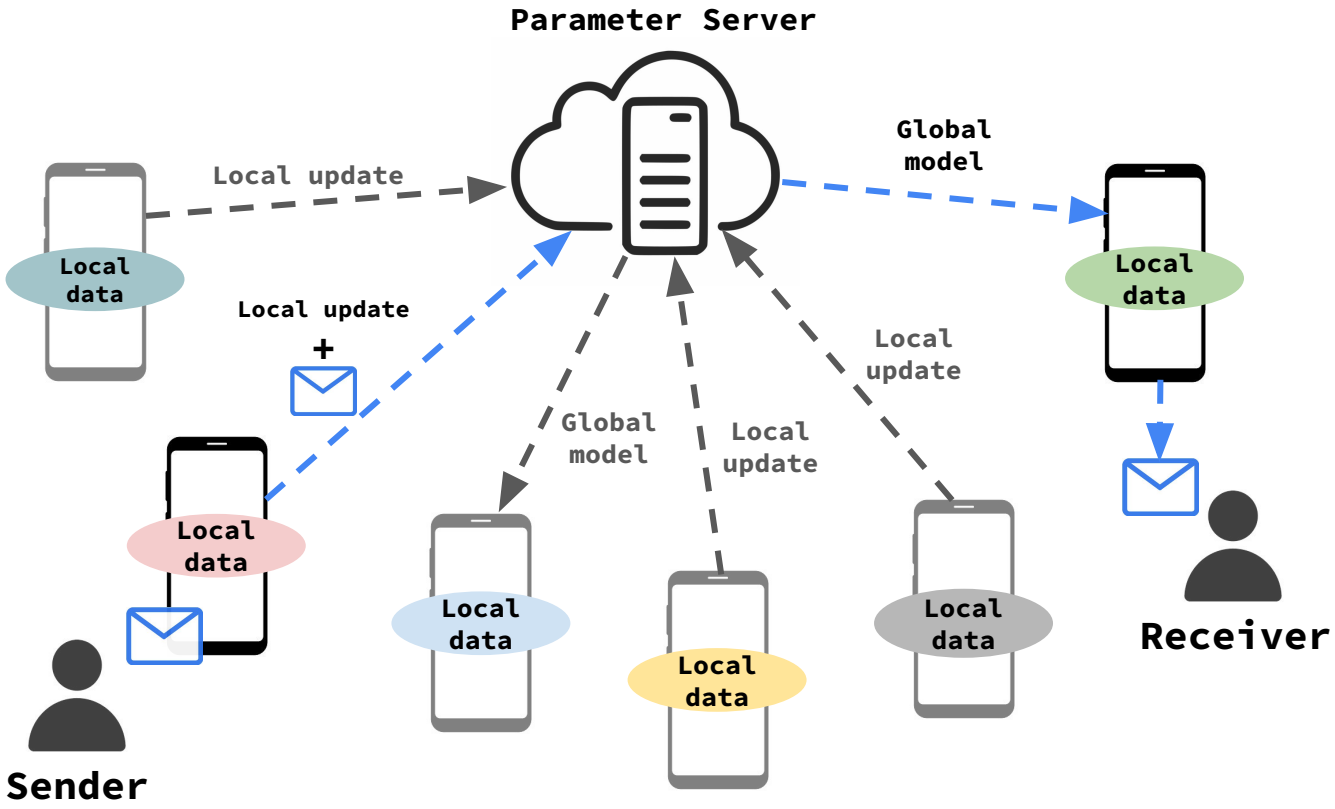


4 senders

100% aggregated per round

- Everything else held equal.

Remarks


Parameter Server
Local update
Global model
Local data
Local update
+
Local data
Global model
Local update
Local update
Local data
Local data
Local data
Local data
Sender
Receiver


That's all Folks!

Reading Material

1. Covert Channels (Concepts and definitions): Link
2. Covert communication in collaborative learning: Link-1, Link-2 (research papers).