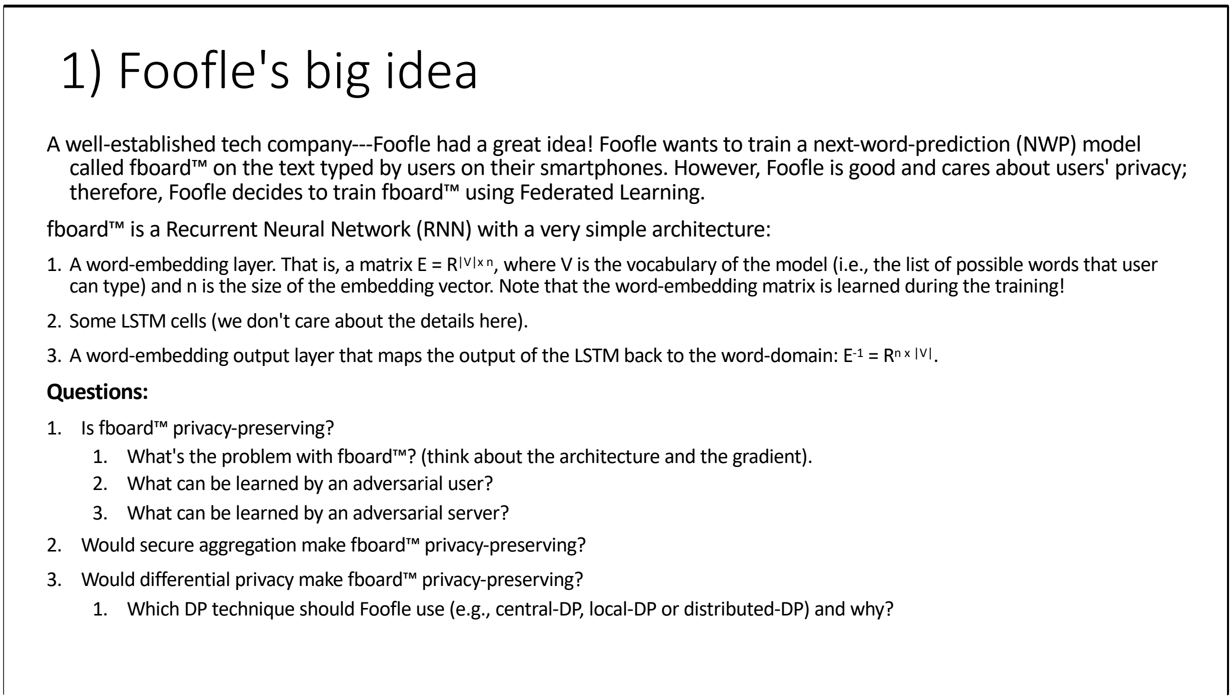
CS-523 Advanced topics on
Privacy Enhancing Technologies

Machine Learning
Live exercises

Carmela Troncoso
SPRING Lab
carmela.troncoso@epfl.ch



1) Foofle's big idea

A well-established tech company---Foofle had a great idea! Foofle wants to train a next-word-prediction (NWP) model called fboard™ on the text typed by users on their smartphones. However, Foofle is good and cares about users' privacy; therefore, Foofle decides to train fboard™ using Federated Learning.

fboard™ is a Recurrent Neural Network (RNN) with a very simple architecture:

1. A word-embedding layer. That is, a matrix E = $R^{|V|\times n}$, where V is the vocabulary of the model (i.e., the list of possible words that user can type) and n is the size of the embedding vector. Note that the word-embedding matrix is learned during the training!
2. Some LSTM cells (we don't care about the details here).
3. A word-embedding output layer that maps the output of the LSTM back to the word-domain: E^{-1} = $R^{n \times |V|}$.

Questions:

1. Is fboard™ privacy-preserving?
    1. What's the problem with fboard™? (think about the architecture and the gradient).
    2. What can be learned by an adversarial user?
    3. What can be learned by an adversarial server?
2. Would secure aggregation make fboard™ privacy-preserving?
3. Would differential privacy make fboard™ privacy-preserving?
    1. Which DP technique should Foofle use (e.g., central-DP, local-DP or distributed-DP) and why?

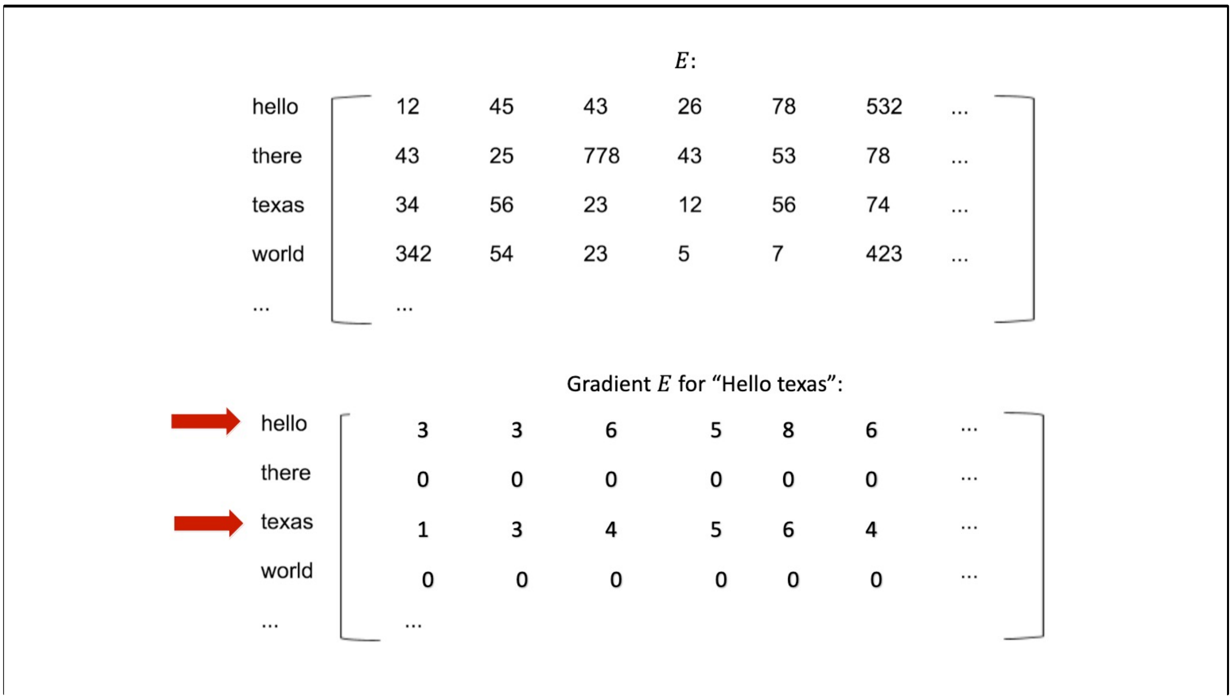
1.1) Embedding layers produce sparse gradient signal!
Only tokens (words) used during the training get non-zero gradient. Check the next slide for visualization.

1.2) Besides the standard information leak from the global model, users can learn something more by just comparing the old global model and the new one (think about the embedding layer).

1.3) Even worse, the server can learn fine-grained information about users by just checking the gradient.

2) Secure aggregation would prevent the server from observing individual gradient signals. Is that enough? The server can still infer precise information about the population.

3.1) Central-DP won’t give any additional privacy guarantees to users against an adversarial server. Instead, distributed-DP can help (if implemented correctly).

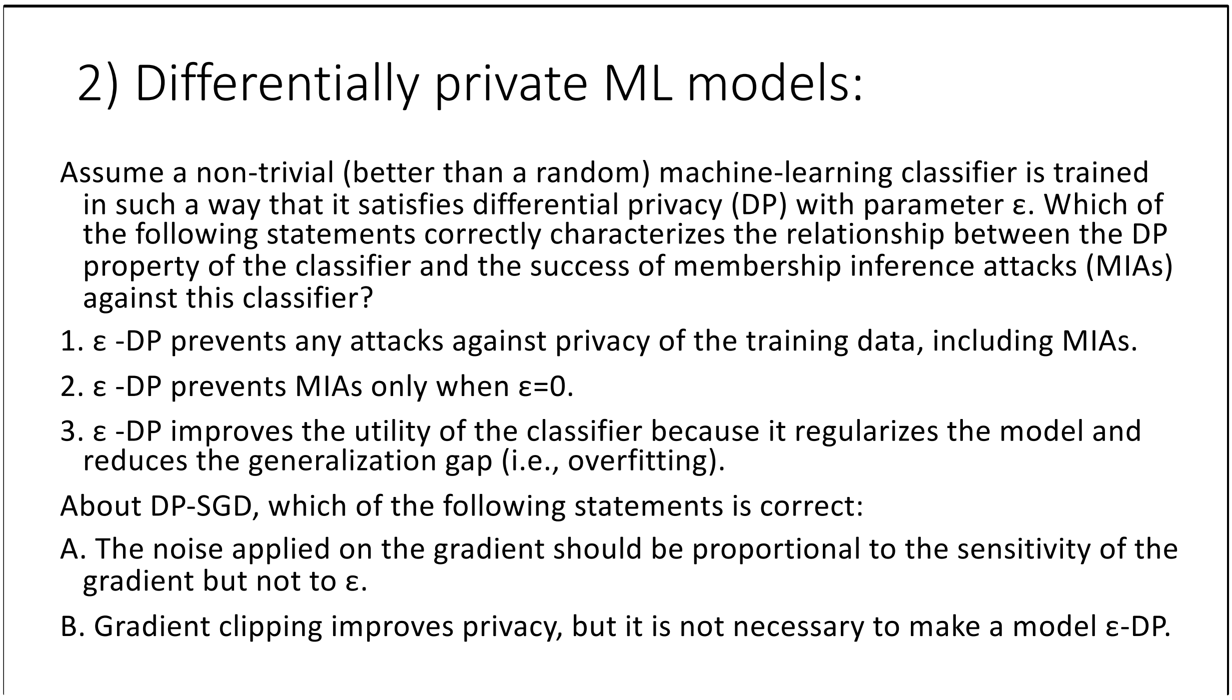


E:

$$\begin{array}{l} \text{hello} \\ \text{there} \\ \text{texas} \\ \text{world} \\ \ldots \end{array} \begin{bmatrix} 12 & 45 & 43 & 26 & 78 & 532 & \ldots \\ 43 & 25 & 778 & 43 & 53 & 78 & \ldots \\ 34 & 56 & 23 & 12 & 56 & 74 & \ldots \\ 342 & 54 & 23 & 5 & 7 & 423 & \ldots \\ \ldots & & & & & & \end{bmatrix}$$

Gradient E for "Hello texas":

$$\begin{array}{rl} \Rightarrow & \text{hello} \\ & \text{there} \\ \Rightarrow & \text{texas} \\ & \text{world} \\ & \ldots \end{array} \begin{bmatrix} 3 & 3 & 6 & 5 & 8 & 6 & \cdots \\ 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\ 1 & 3 & 4 & 5 & 6 & 4 & \cdots \\ 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\ \ldots & & & & & & \end{bmatrix}$$



2) Differentially private ML models:

Assume a non-trivial (better than a random) machine-learning classifier is trained in such a way that it satisfies differential privacy (DP) with parameter ε. Which of the following statements correctly characterizes the relationship between the DP property of the classifier and the success of membership inference attacks (MIAs) against this classifier?

1. ε -DP prevents any attacks against privacy of the training data, including MIAs.
2. ε -DP prevents MIAs only when ε=0.
3. ε -DP improves the utility of the classifier because it regularizes the model and reduces the generalization gap (i.e., overfitting).

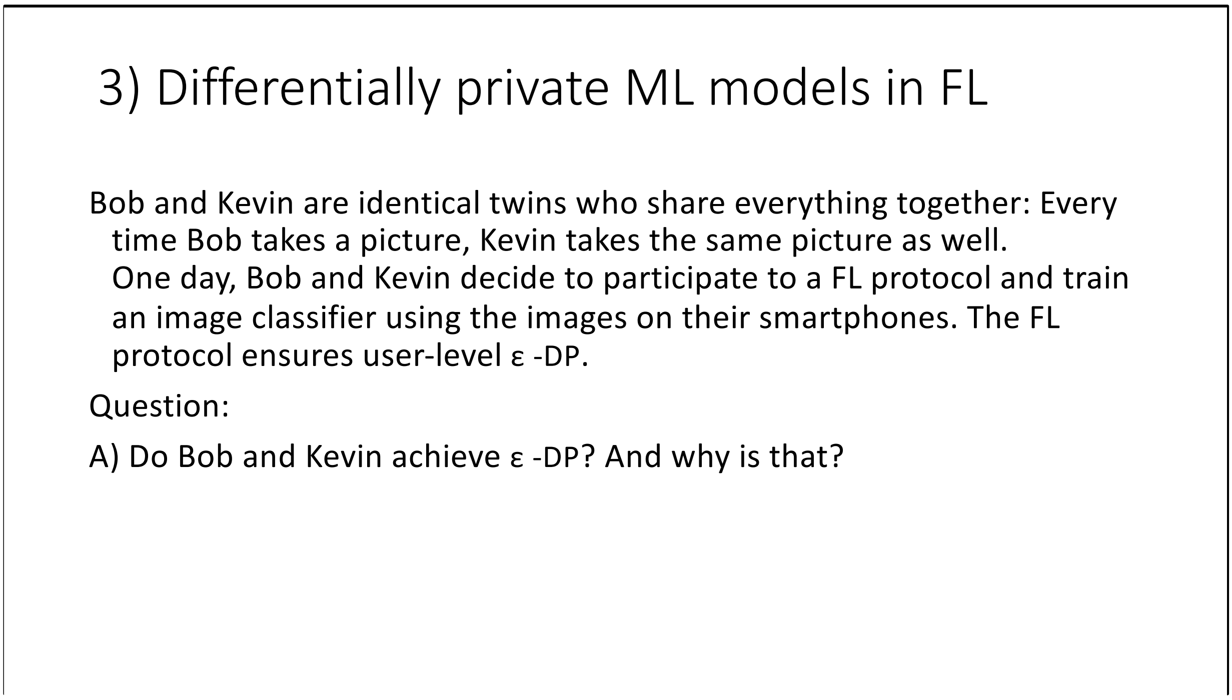
About DP-SGD, which of the following statements is correct:

A. The noise applied on the gradient should be proportional to the sensitivity of the gradient but not to ε.

B. Gradient clipping improves privacy, but it is not necessary to make a model ε-DP.

1) The differential privacy definition does not cover property inference attacks. That is, inference attacks over general statistics of the training set would work.
2) When \epsilon=0, the model does not store information about the training set and, therefore, cannot leak information about it. When \epsilon>0, the model is vulnerable to privacy attacks but the accuracy is upper bounded by a quantity dependent on \epsilon. The smaller the \epsilon, the lower the upper bound.
3) DP does reduce overfitting, but it achieves this by destroying the useful information provided by the training set —> i.e., worse performance for the model.

A) The noise must be proportional to both sensitivity and inversely proportional to the desired privacy budget.
B) Clipping is a fundamental operation as this bounds the sensitivity of the gradient (which would be unbounded otherwise) and allows us to account for the worst-case-scenario.



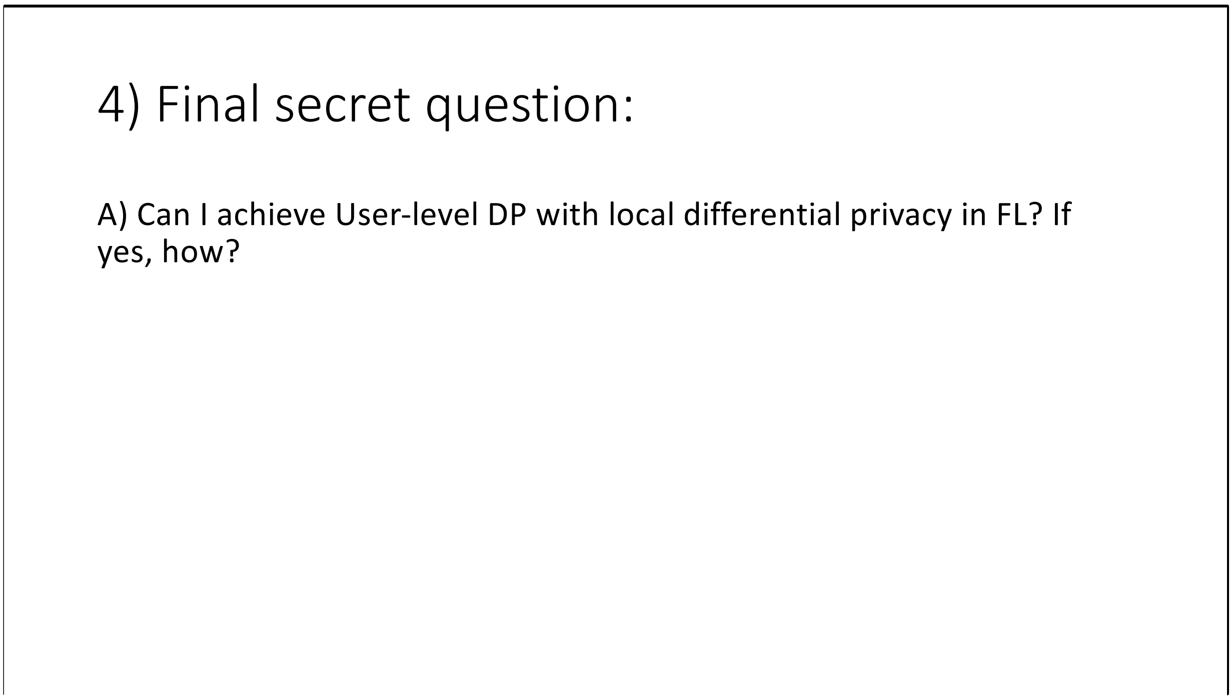
3) Differentially private ML models in FL

Bob and Kevin are identical twins who share everything together: Every time Bob takes a picture, Kevin takes the same picture as well. One day, Bob and Kevin decide to participate to a FL protocol and train an image classifier using the images on their smartphones. The FL protocol ensures user-level ε -DP.

Question:

A) Do Bob and Kevin achieve ε -DP? And why is that?

No! Essentially, Bob and Kevin are participating twice to the learning protocol. This implies that their privacy loss is increasing faster compared to the other users. Eventually, at the end of the training, they would achieve $\epsilon' - DP$, where $\epsilon' > \epsilon$ (which means less privacy).



4) Final secret question:

A) Can I achieve User-level DP with local differential privacy in FL? If yes, how?

Yes, by averaging the gradient computed locally before clipping and noise addition.

However, the signal-to-noise ration would be really bad and nearly all the information in the gradient would be destroyed