Qualcomm

Federated Learning of User verification Models Without Sharing Embeddings

Hossein Hosseini

Hyunsin Park, Sungrack Yun, Christos Louizos, Joseph Soriaga, Max Welling

Federated Learning (FL)

- **Private:** raw data stays on device.

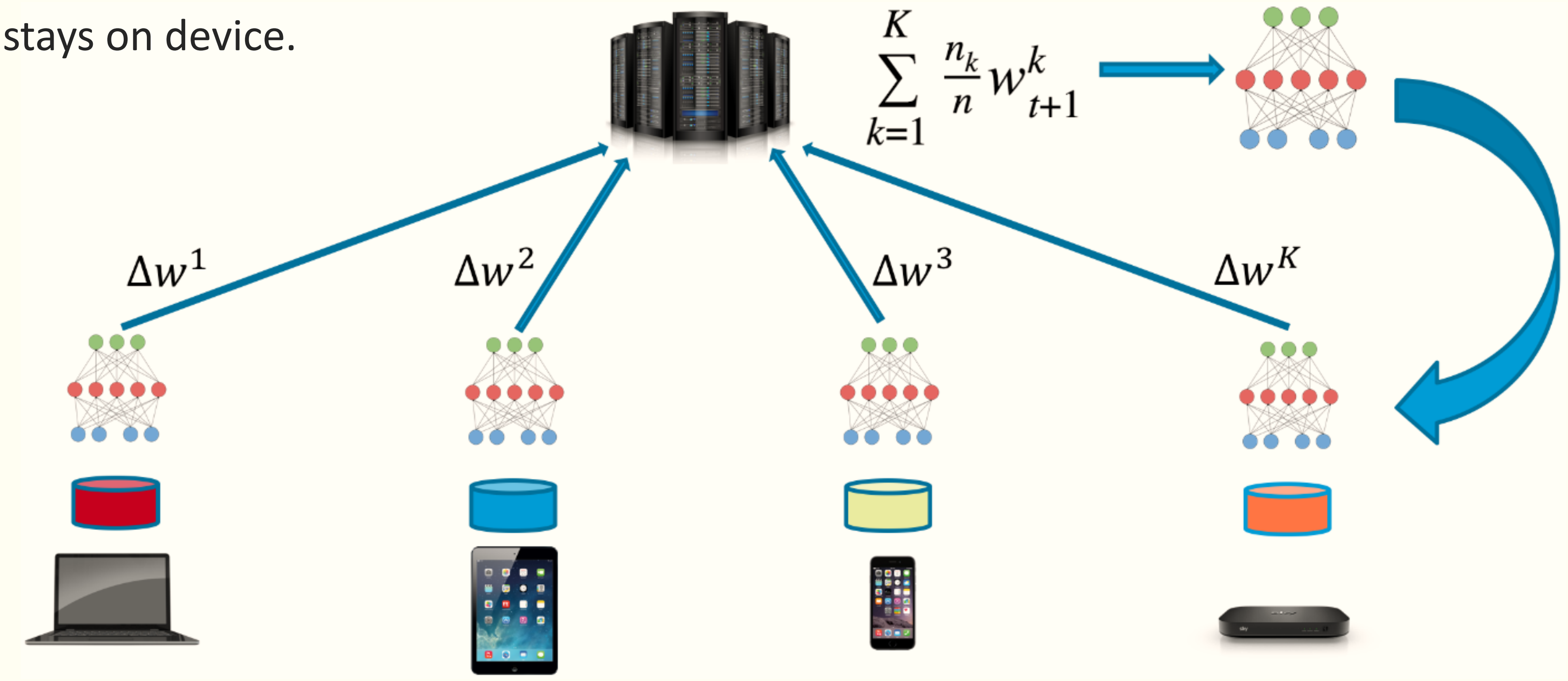


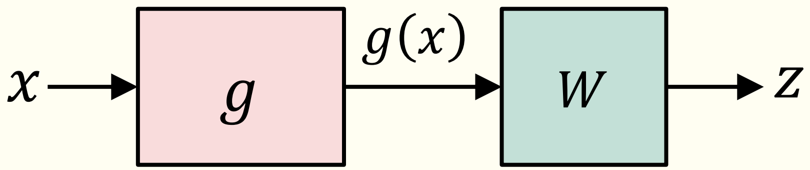
Figure from: https://proandroiddev.com/federated-learning-e79e054c33ef

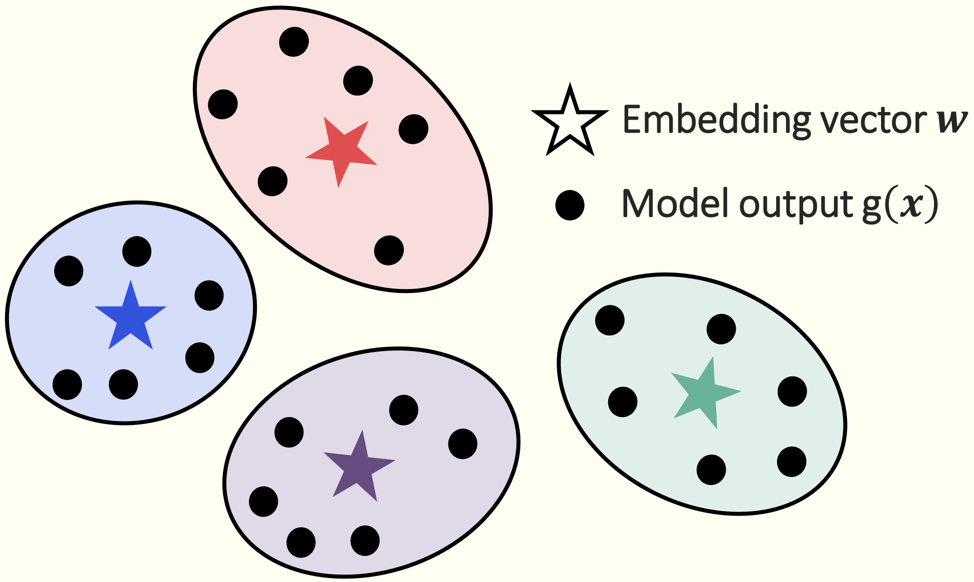
User Verification (UV)

- User verification is the task of accepting/rejecting users based on their input data.
  - Usually done using some biometric data such as face, voice, fingerprint, etc.
  - Deployed on edge devices for unlocking the device or providing specific services.

User Verification Models- Training and Inference

- UV with machine learning:
  - Cluster users' data in embedding space, s.t. embedding of data of each user is:
    - Close to the embedding vector of that user,
    - Far away from embedding vectors of other users.





- **Training loss function:** $\ell = l_{\text{pos}} + \lambda l_{\text{neg}}$
  - $l_{\text{pos}} = d\big(g(x), w_y\big)$ → minimizes distance of $g(x)$ to embedding vector of corresponding user.
  - $l_{\text{neg}} = -\min_{u \neq y} d\big(g(x), w_u\big)$ → maximizes distance to embedding vectors of other users.

Challenges of Training UV Models

- Data collection:
  - UV models need to be trained with large and diverse data for best performance.
  - Collecting data centrally not feasible due to privacy constraints of raw biometric inputs.
  - **Use federated learning:** FL enables training without having direct access to data.
- How about embeddings?
  - Embeddings are used for verifying users, hence are security-sensitive info and cannot be shared with server or other users
  - $\Longrightarrow$ users cannot compute $l_{\text{neg}} = -\min_{u \neq y} d(g(x), w_u)$
  - Training with only $l_{\text{pos}} = d\big(g(x), w_y\big)$ causes all embeddings to collapse into same vector (loss will be 0).

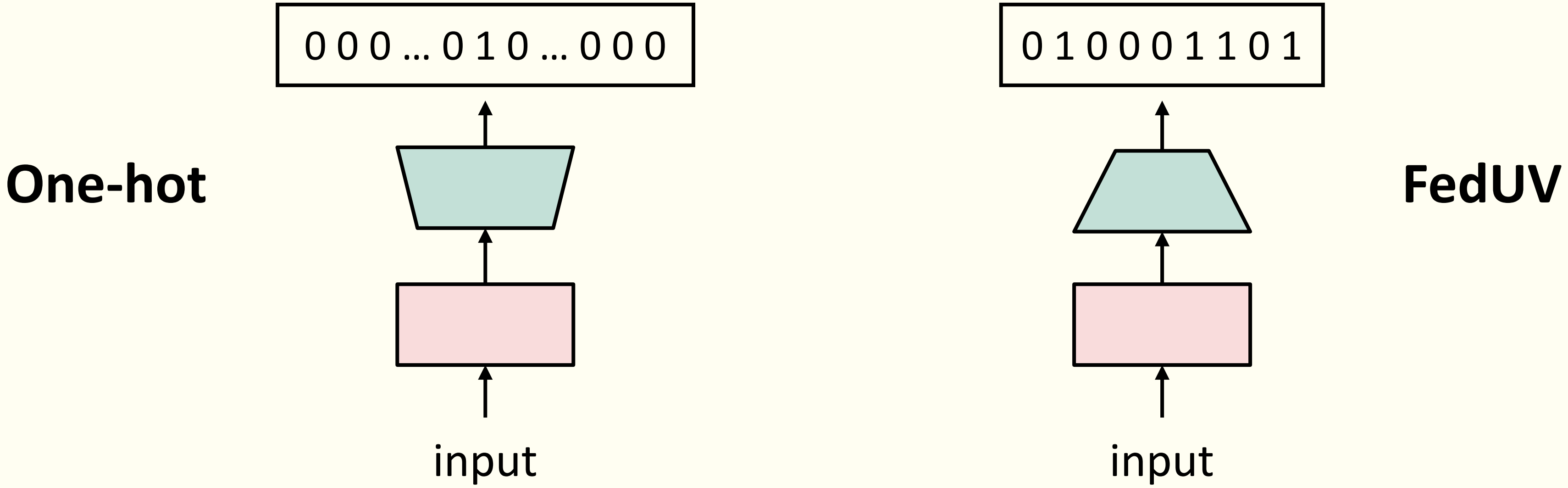
Related Work: Federated Averaging with Spreadout (FedAwS), [ICML '20]

- **Theorem:** higher min distance between embeddings → higher classification accuracy.
  - What we want: train with l_{pos} and ensure w_i's are highly separable.

- Original loss function: $\ell(x, y; g, w) = d\big(g(x), w_y\big) - \lambda \sum_{u \neq y} d\big(g(x), w_u\big)$
- FedAwS loss function: $\ell(x, y; g, w) = \underbrace{d\big(g(x), w_y\big)}_{\text{done by users}} - \lambda \underbrace{\sum_{u \neq y} d\big(w_y, w_u\big)}_{\text{done by server}}$
  - **Theorem:** positive loss + spreadout loss ∼ original loss.

- **Problem:** embedding of each user is kept private from other users but not from **server**.

Proposed Method: Federated User Verification (FedUV)

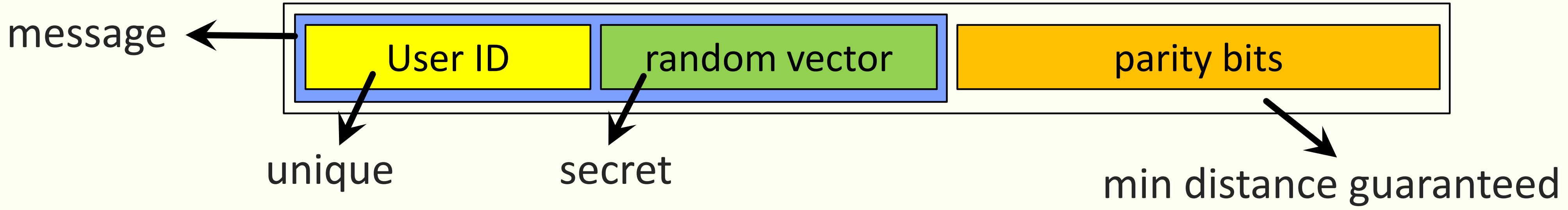


- Users jointly learn a set of vectors (W), but each user minimizes distance of $g(x)$ to a secret linear combination (v) of those vectors.
  - Original loss function: $\ell(x, y; g, w) = d\left(g(x), w_y\right) - \lambda \min_{u \neq y} d\left(g(x), w_u\right)$
  - **FedUV** loss function: $\ell(x, y; g, w) = d\left(g(x), W^T v_y\right) - \lambda \min_{u \neq y} d\left(g(x), W^T v_u\right)$

Error-correcting Codes (ECCs) Codewords as Secret Vectors

- **Theorem:** With v_u's chosen from ECC codewords, minimizing l_{pos} also minimizes l_{neg}.
  - Hence, negative loss term becomes redundant.

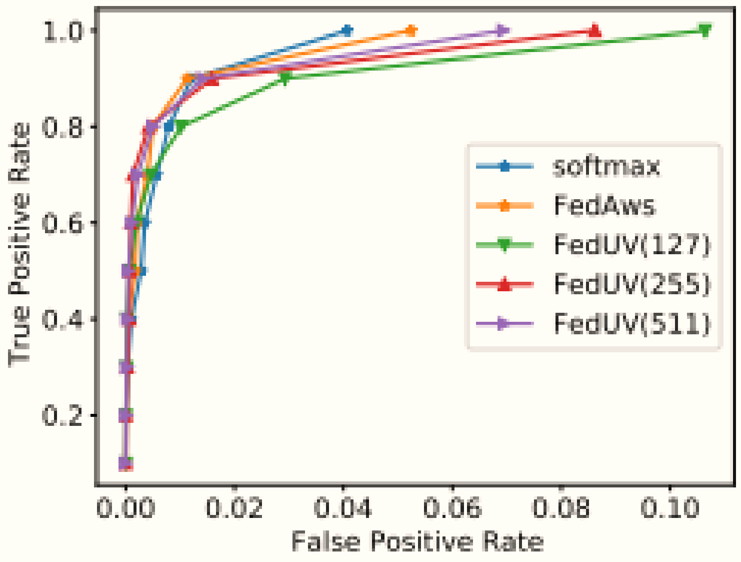
- How to construct secret codewords?



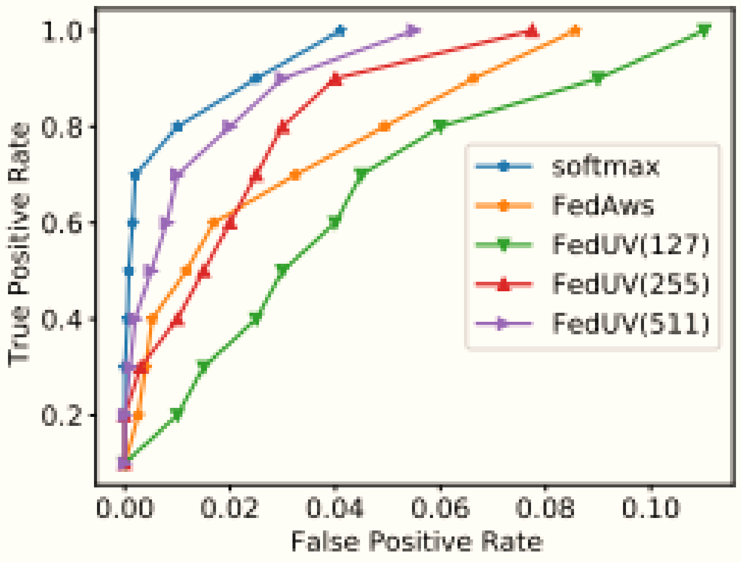
  - **Properties:**
    - Vectors are **unique** because the user ID is unique,
    - Vectors are **secret** because the random vector is not known to other users or the server,
    - Vectors are guaranteed to be **maximally separated** due to the use of ECC algorithms.

Experimental Results- UV with voice, face and handwriting data

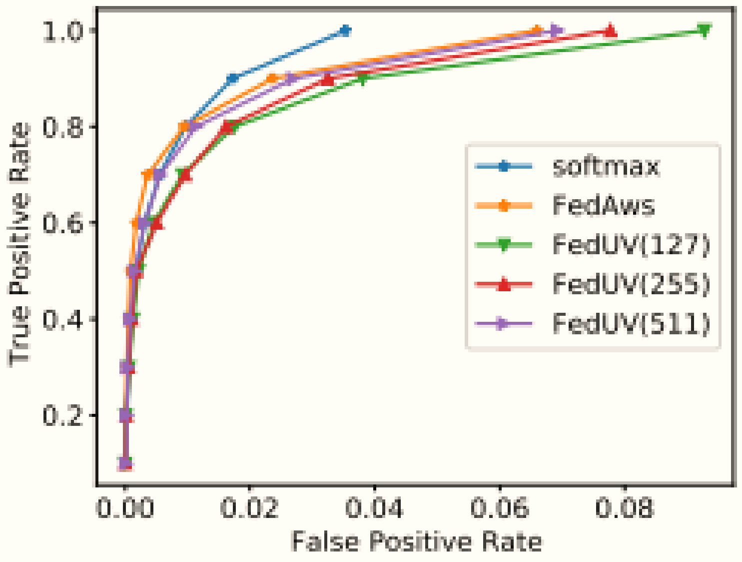
- Settings: 1000 users, BCH code for generating codewords.
- Methods:
  - **Baselines:** softmax (regular federated learning with one-hot encoding) and FedAWS [Yu et al., ICML ‘21].
  - **Our method:** FedUV(c) denotes FedUV with code length of c.
- FedUV on par with existing approaches, without sharing embeddings with other users or server.



Voice data (VoxCeleb)



Face data (CelebA)



Handwriting (MNIST-UV)

Thank You!

Paper: https://arxiv.org/abs/2104.08776

Hossein Hosseini hhossein@qti.qualcomm.com