Mitigating the Impact of Federated Learning on Client Resources

Sebastian Caldas[1], Jakub Konečný[2], H. Brendan McMahan[2], Ameet Talwalkar[1,3]

[1]Carnegie Mellon University, [2]Google, [3]Determined AI

Contributions

- We address the challenge of bringing Federated Learning (FL) to realistic heterogeneous edge networks.
- To address communication bottlenecks, we use lossy compression on the exchanges sent from server-to-client and client-to-server.
- To prevent computation bottlenecks, we locally train *Federated Submodels,* smaller subsets of the full global model.

Federated Submodels

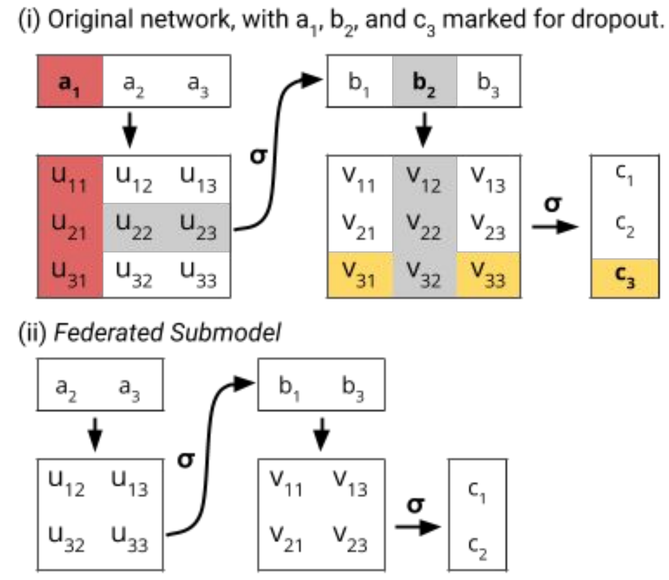

Figure 2: The *Federated Submodels* strategy applied to two fully-connected layers. Submodels are created by dropping out activations.

Each client locally trains an update to a submodel instead of the global model.

- This strategy reduces both the number of FLOPS per gradient evaluation and the model's communication footprint.

End-to-End Strategy

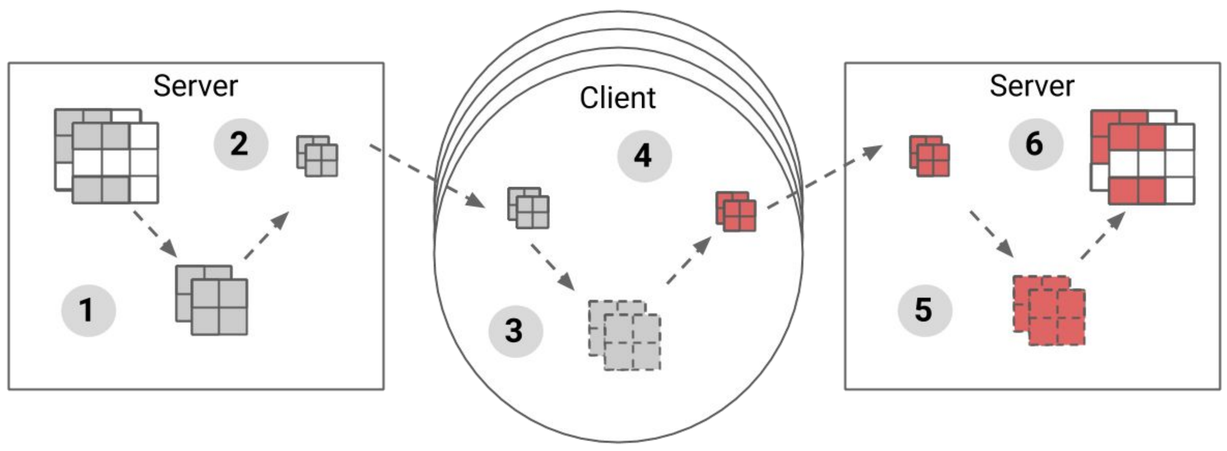

Figure 1: Combination of our proposed strategies. We start by (1) constructing a *Federated Submodel*, and by (2) lossily compressing the resulting object. This compressed model is then sent to the client, who (3) decompresses, trains it using local data, and (4) compresses the final update. This update is sent back to the server, where it is (5) decompressed and, finally, (6) aggregated into the full global model.

Lossy Compression

- Our techniques are built upon those successfully used by [1] to compress client-to-server updates. However, we also apply them to server-to-client downloads.
- We proceed in four steps:
  - Reshape each weight matrix into a vector.
  - Apply a **basis transform** to the vector.
  - **Subsample** a *s*-fraction of its elements.
  - **Quantize** the remaining elements to *q* bits.

Kashin's Representation

- Kashin's representation [2] spreads a vector's information *as much as possible* in every dimension.
- This representation further mitigates the error incurred by subsequent quantization compared to using the random Hadamard transform (as in [1]).

References:

[1] J. Konečný et al. Federated learning: Strategies for improving communication efficiency. arXiv:1610.05492, 2016.

[2] Y. Lyubarskii and R. Vershynin. Uncertainty principles and vector quantization. IEEE Transactions on Information Theory, 2010.

Experimental Results

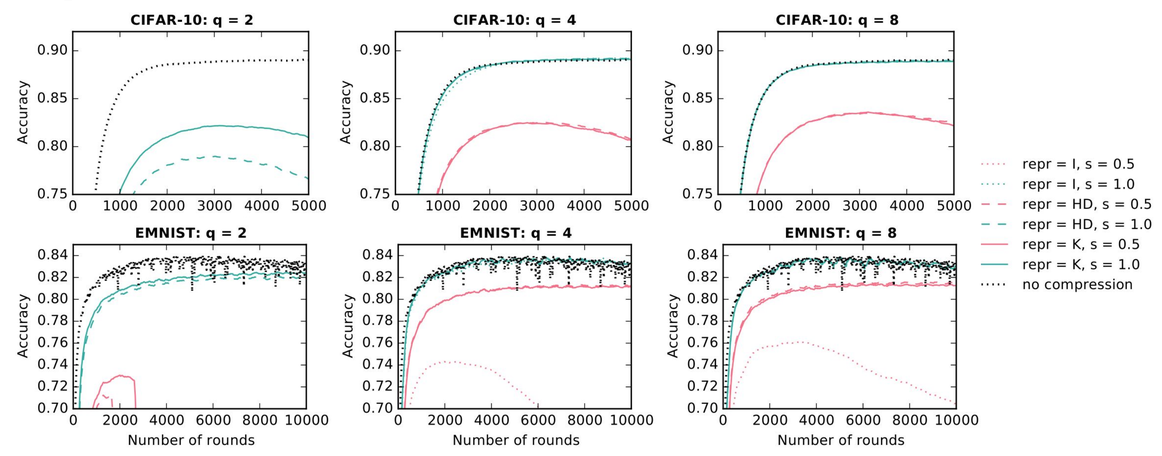

Figure 3: Effect of lossy compression. We vary the type of basis transform (Identity, Hadamard and Kashin's), the subsampling rate *s* and the number of quantization bits *q*. **We can match our *no compression* baseline using *q*=4, which amounts to a 8x reduction in communication.**

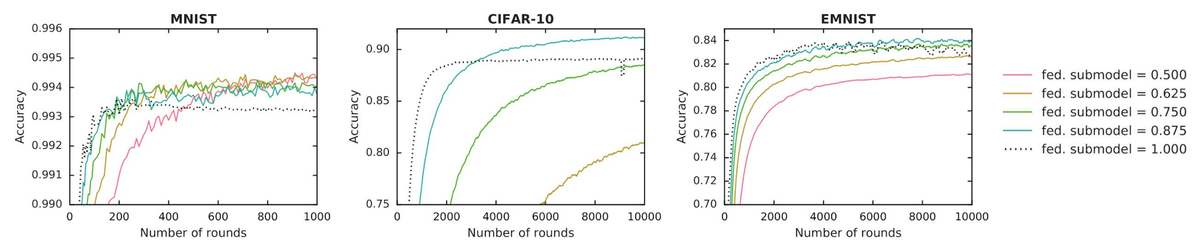

Figure 4: Results for *Federated Submodels* when varying the percentage of neurons kept. **We can get up to ~43% (cor. 25%) savings in fully-connected layers (cor. conv).**

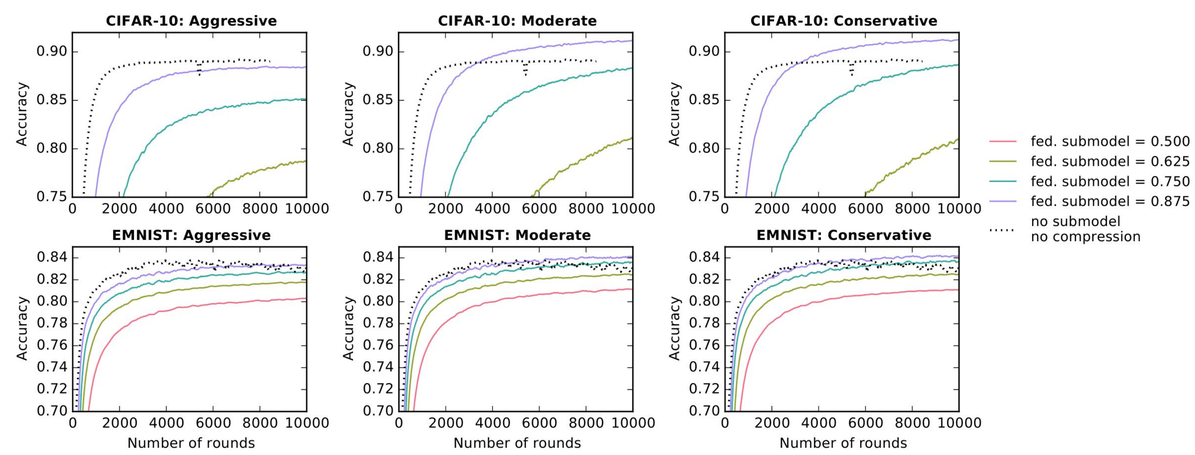

Figure 5: Effects of our end-to-end strategy under 3 empirically chosen compression schemes. **It allows for up to a 14x reduction in server-to-client communication, a 1.7x reduction in local computation and a 28x reduction in client-to-server communication.**