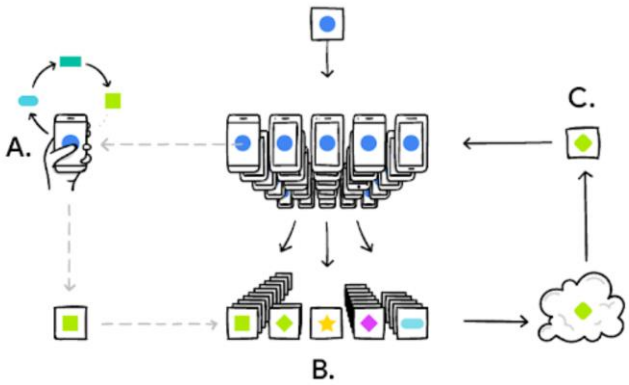
Federal study: Introduction to patterns, technologies and applications

Background

- What is Federal study?

Privacy-protected multi-body, multi-agency collaborative machine learning model
Raw data stays in local
equal status, sharing of results

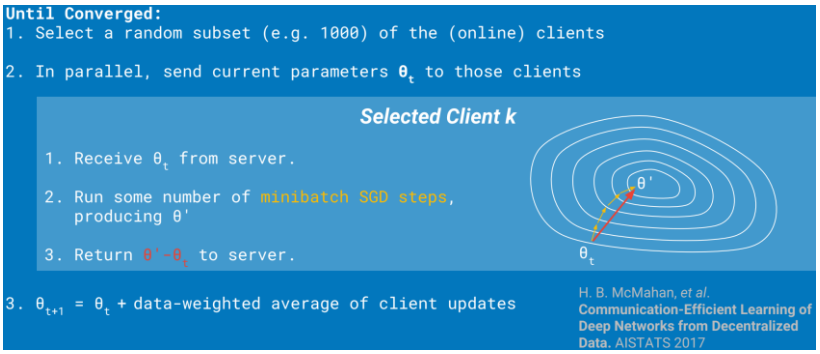


- Why federal study?

Because of the phenomenon of data island.
Individuals and institutions have accumulated more and more data, but these data are divided and not circulated. This is the data island, and each organization will only have its own data. The same thing happens on machine learning areas.
If the federal learning can solve the problem of data privacy security, then what is the benefit of federal learning for the construction of machine learning models?
The challenge that federal learning needs to solve is the data privacy security issue, which is the main reason of data island. The consideration of privacy protection is also a place where the federation differs from general collaborative machine learning and distributed machine learning.

Federated stochastic gradient descent FedAvg

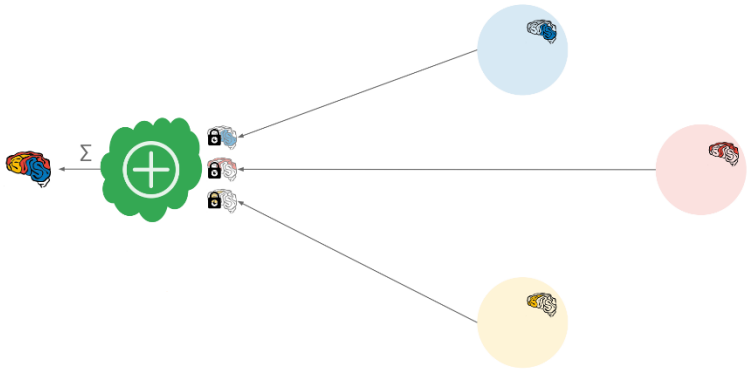
Method: Select some terminals to update the model in the cloud Average. Output training updated model parameters on local data. Fusion model in the cloud (collaborative learning).
Risk: Gradient is calculated from raw data (such as text sequences), multiple iterations will increased the risk



How to avoid risks:

- Multiparty secure computing: Avoid individual gradient leaks

Each user encrypts its own model parameter variation as plaintext, and then sends the ciphertext to the server. The addition of homomorphic encryption ensures that the server can only see changes in model parameters of all users. The sum of the quantities, and basically can not be inferred from the metrics of the parameter variation of each user. This avoids the server directly observing the variation of model parameters of each user, reducing the risk of privacy leakage.



- Differential privacy: stricter data privacy guarantee

We will assume that the server side and other users are untrustworthy. After getting the variation of our model parameters, I will add some noise first, and then encrypt and transmit it to the server. In this case, even if the server side is combined with other users for privacy attacks, it is impossible to get the amount of changes in my real model parameters, and I can't get the value of my real sample. Here, to achieve differential privacy, we need to add Laplace noise to the gradient, and add Gaussian noise to get approximate differential privacy. The greater the amount of noise added, the better the privacy protection, but the slower the convergence of the model, which requires a trade-off balance.