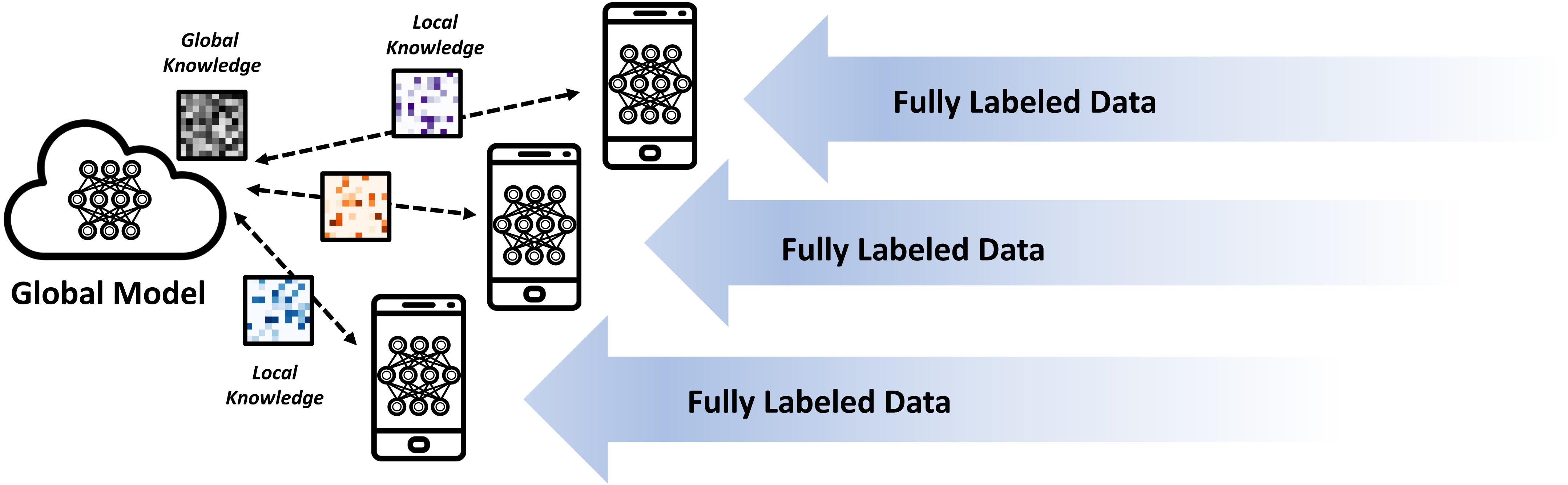
Federated Semi-Supervised Learning with Inter-Client Consistency & Disjoint Learning

Wonyong Jeong[1], Jaehong Yoon[2], Eunho Yang[1,3], Sung Ju Hwang[1,3]

Graduate Shool of AI[1], KAIST, Seoul, South Korea
School of Computing[2], KAIST, Deajeon, South Korea
AITRICS[3], Seoul, South Korea

Motivation: Deficiency of Supervision

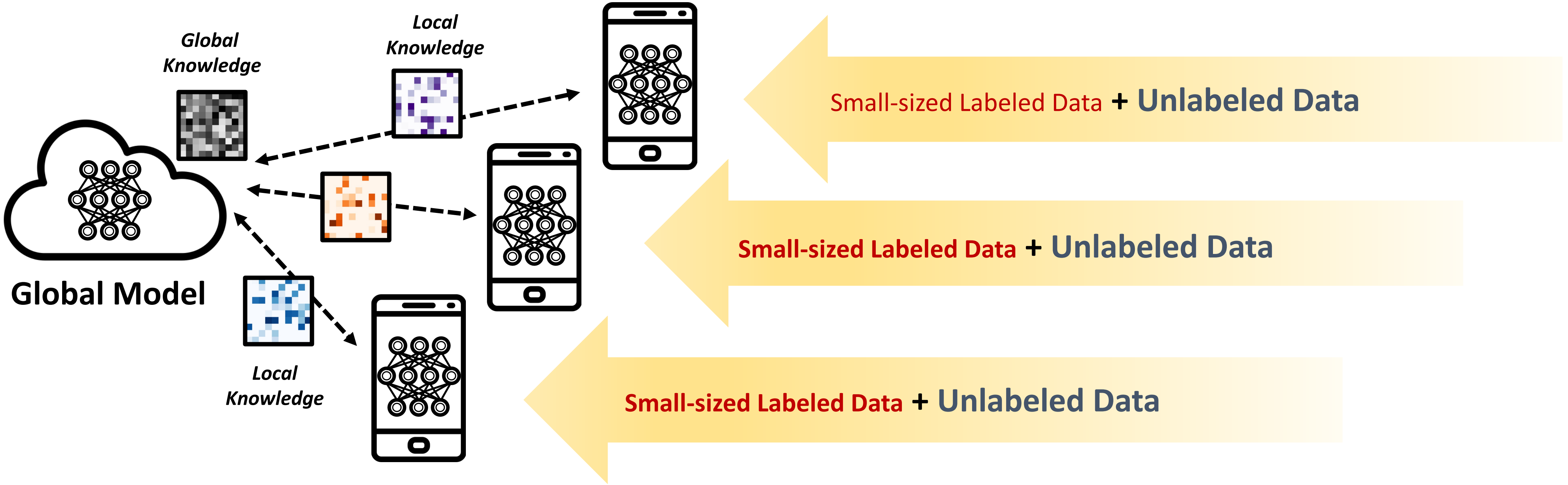
Federated Learning, in which multiple clients collaboratively learn a global model via aggregating knowledge from local private data, have been actively studied.



Data obtained at the client often comes without accompanying labels due to **expensive labeling costs** or **requirement of expert knowledge** when annotating.

Federated Semi-Supervised Learning

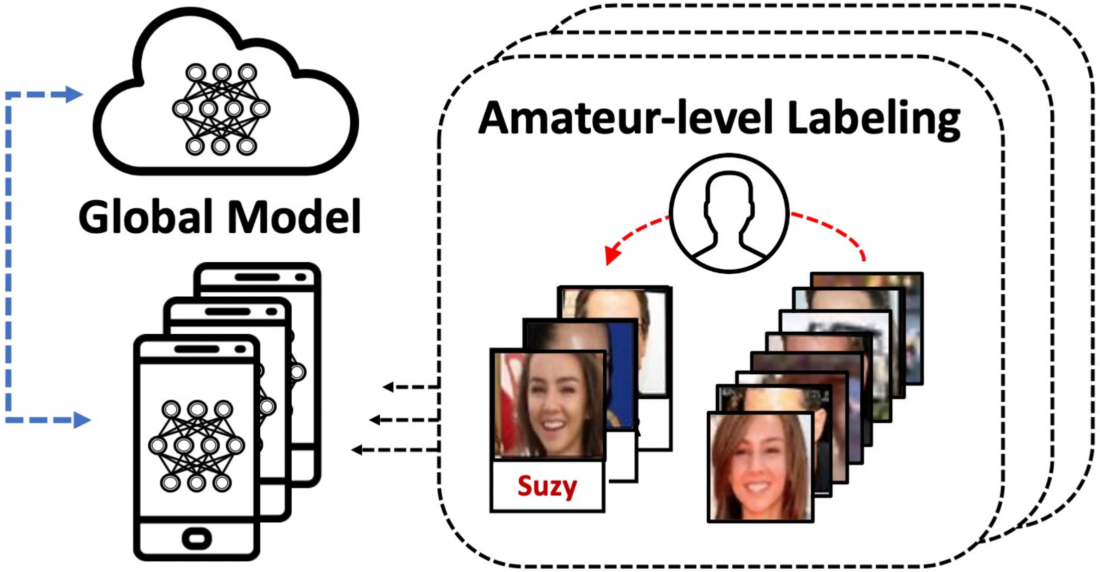
This leads us to a new problem of **Federated Semi-Supervised Learning** (FSSL), which tackles federated learning under scarcity of supervision.



We propose two possible scenarios depending on the availability of the labeled data, such as **Labels-at-Client** and **Labels-at-Server** scenarios.

Federated Semi-Supervised Learning

One of common scenarios is that the end-users intermittently annotate a small portion of their local data, while the rest of data instances remains unlabeled.



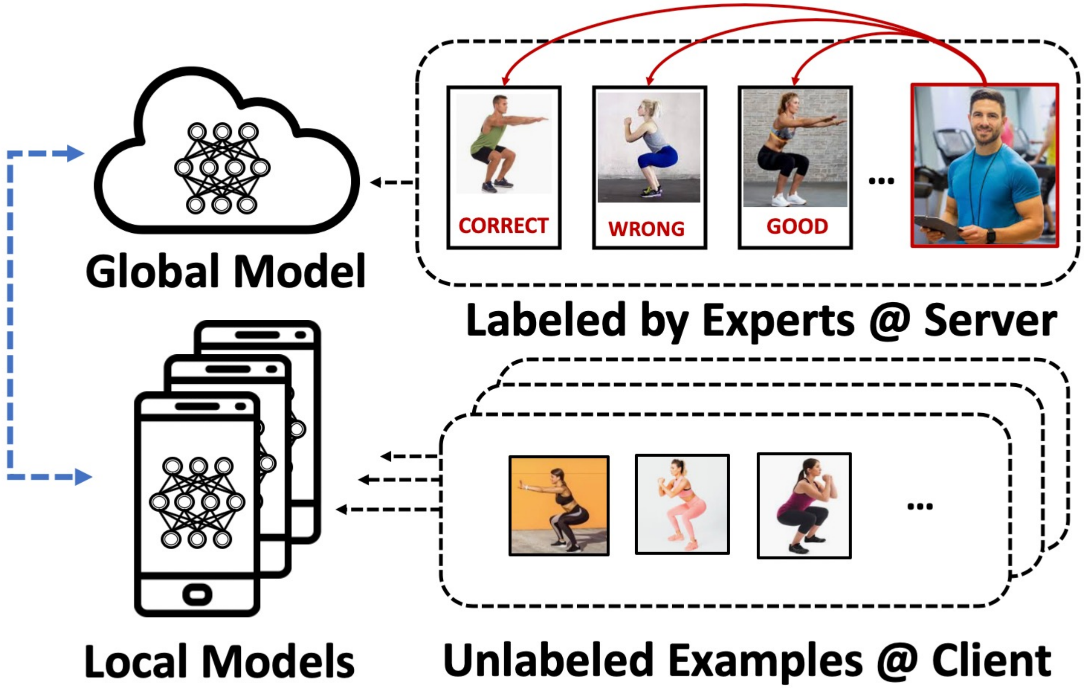
Local Models **Labeled & Unlabeled @ Client**

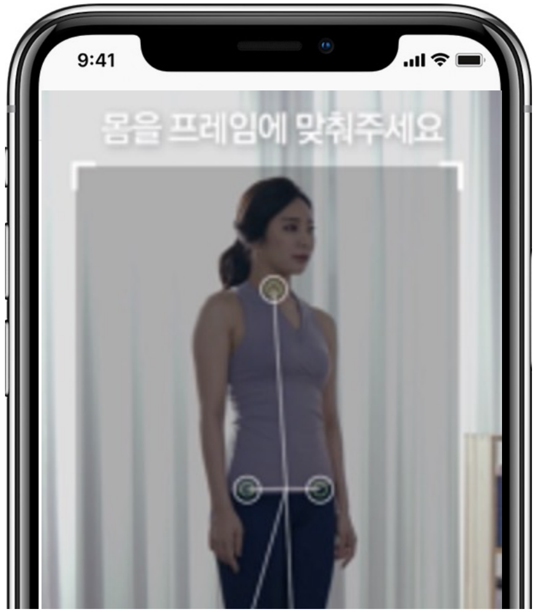
iPhone Photos – Face Album

We call such scenarios as **Labels-at-Client** scenario where both the labeled and unlabeled data are available at client side.

Federated Semi-Supervised Learning

Another common cases for real-world applications is that annotation requires expert knowledge (e.g. annotating medical images, evaluating body postures for exercises).



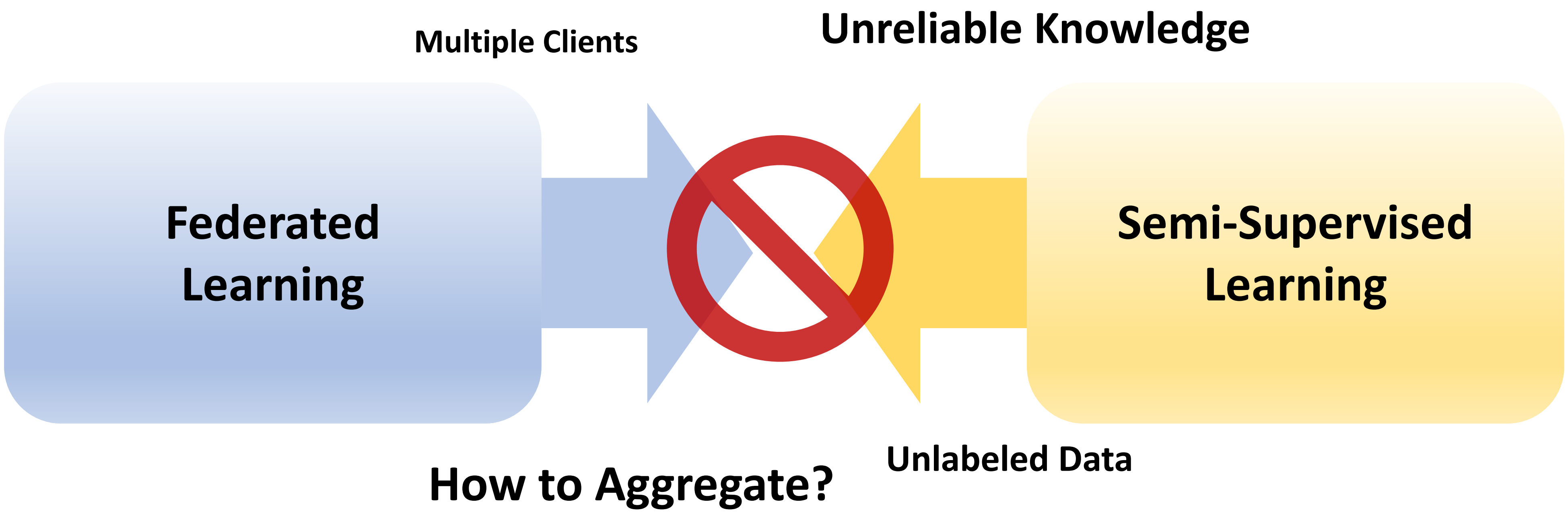


Evaluating Body Postures

We define the scenario as **Labels-at-Server** scenario that the supervised labels are only available at the server, while local clients work with unlabeled data.

Challenges: Exploiting Reliable Knowledge

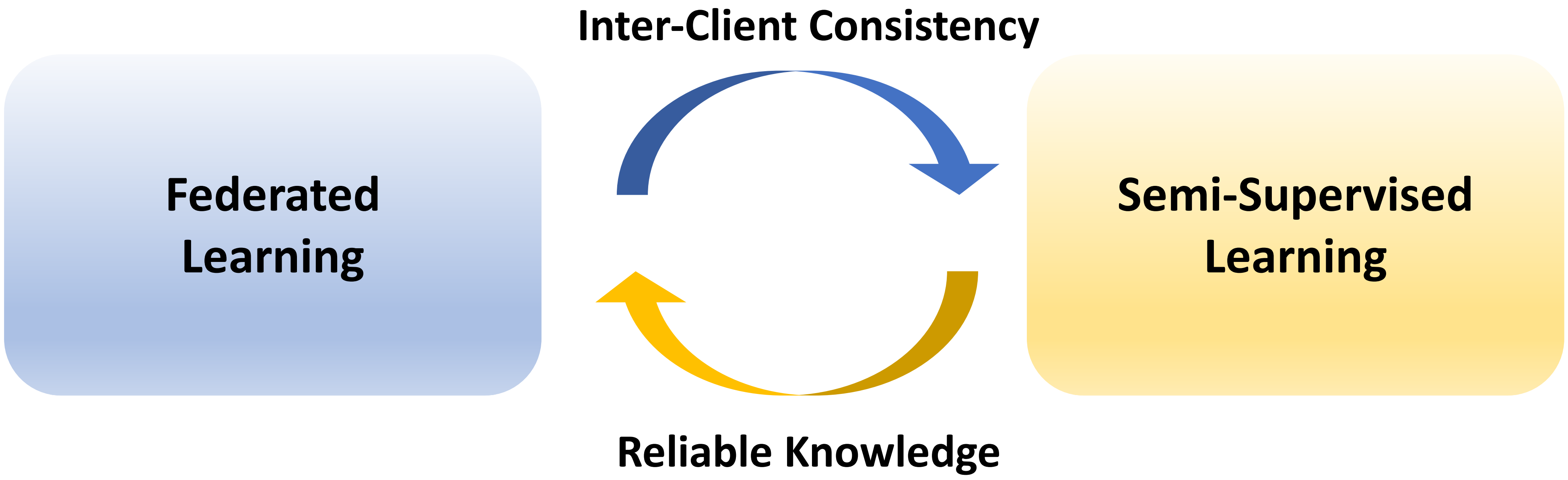
A simple solution to tackle the FSSL problems would be a naïve combination of semi-supervised learning and federated learning algorithms.



Yet, the naïve approach does not fully exploit the knowledge from the multiple models trained on each local unlabeled data, which may cause **the degradation of reliability**.

Federated Matching (FedMatch)

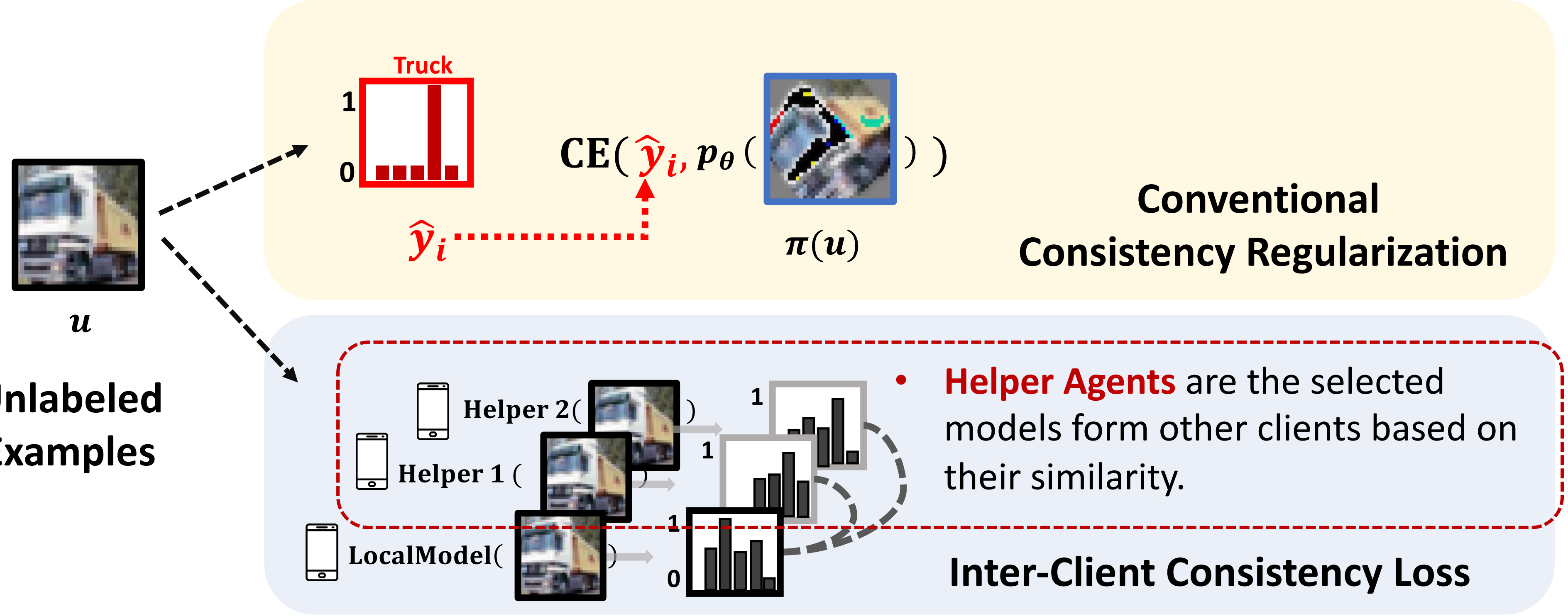
Thus, we propose a novel method, **FedMatch**, which effectively utilizes reliable knowledge from multiple clients.



FedMatch consists of two main components, such as the **Inter-Client Consistency Loss** and **Parameter Decomposition** for disjoint learning.

Inter-Client Consistency

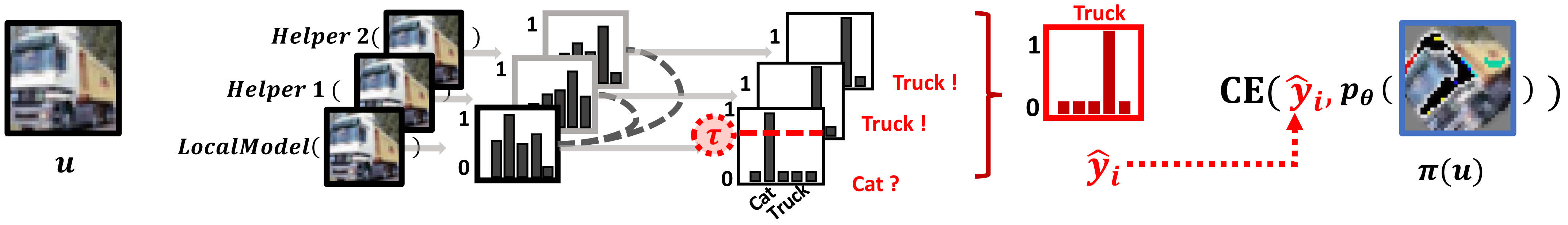
Consistency regularization methods, one of popular approaches in SSL, enforce model predictions from augmented instances to output the same class label.



We further improve this by proposing **Inter-Client Consistency Loss**, which enforces the consistency between the predictions made across multiple helper agents.

Inter-Client Consistency

Moreover, we perform local consistency regularization while utilizing the agreement for generating pseudo labels, namely **agreement-based pseudo labeling**.

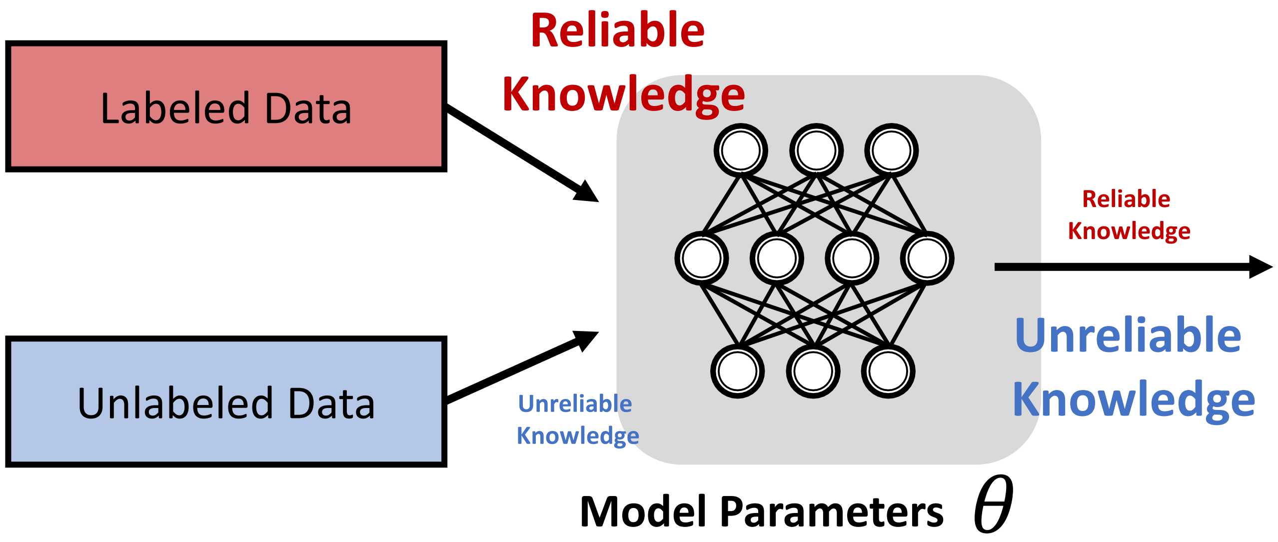


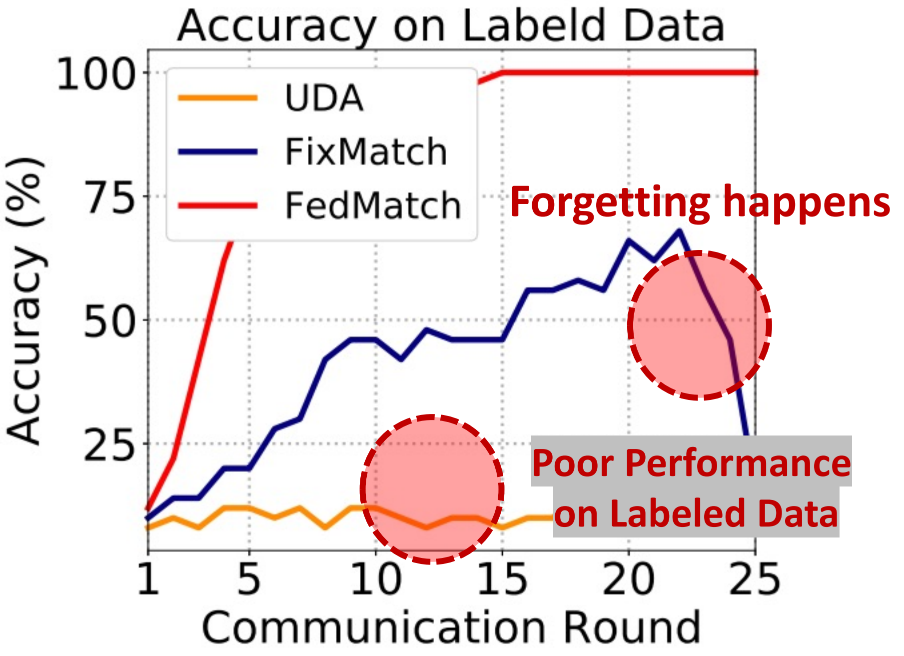
Unlabeled Examples | **Inter-Client Consistency Regularization** | **Agreement-based Pseudo-Labeling** | **Augmentation & Local Consistency Loss**

With the pseudo labels, we minimize standard cross entropy on augmented instances, similarly to FixMatch's (Sohn et al., 2020).

Parameter Decomposition for Disjoint Learning

In the standard semi-supervised learning, learning on labeled and unlabeled data is **simultaneously** done with a shared set of parameters.

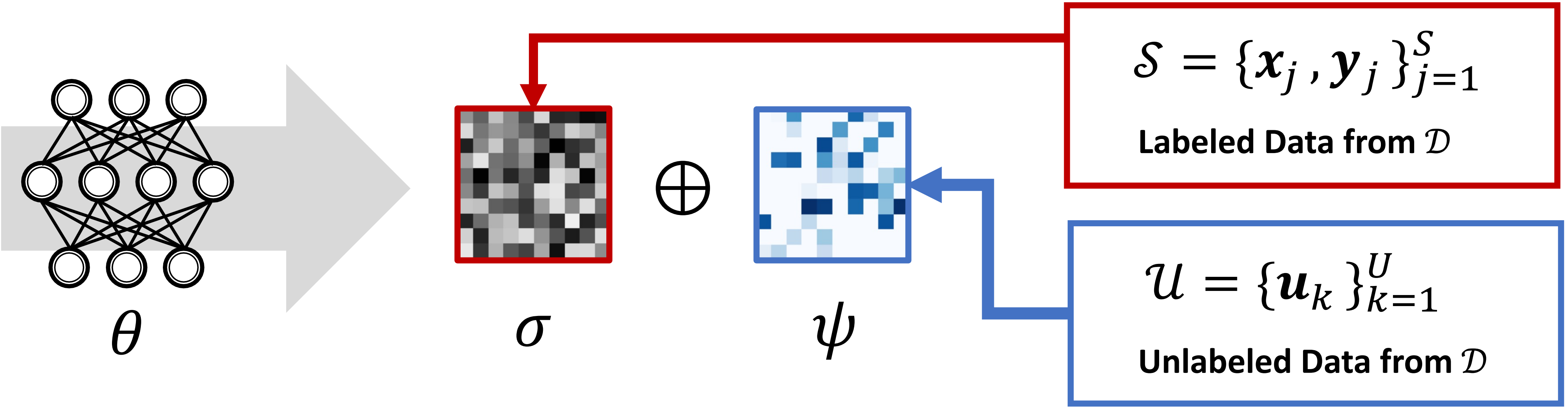




However, this may result in the model to forget about what it learned from labeled data, which is crucial to semi-supervised learning.

Parameter Decomposition for Disjoint Learning

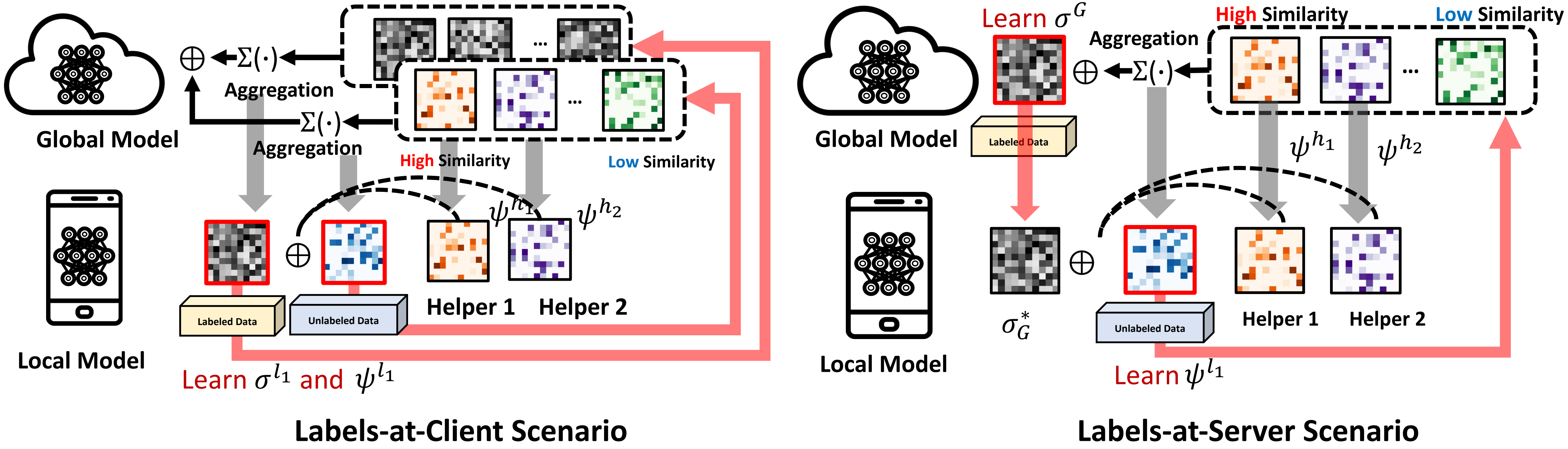
We thus **decompose** our model parameters into parameters for supervised learning and unsupervised learning, which are σ and ψ, respectively.



This enables us to **separate the learning procedures** depending on the availability of labeled data.

Parameter Decomposition for Disjoint Learning

For **Labels-at-Client** scenario, where the labeled data is given at client, we learn both σ and ψ at client side, and send them back to server.



For **Labels-at-Server** scenario, where the labeled data is located at server, we disjointly learn σ at server. In this case, only ψ at client side is sent back to server.

Objective Functions

For learning on labeled data, we perform the standard supervised learning with cross-entropy loss utilizing corresponding labels .

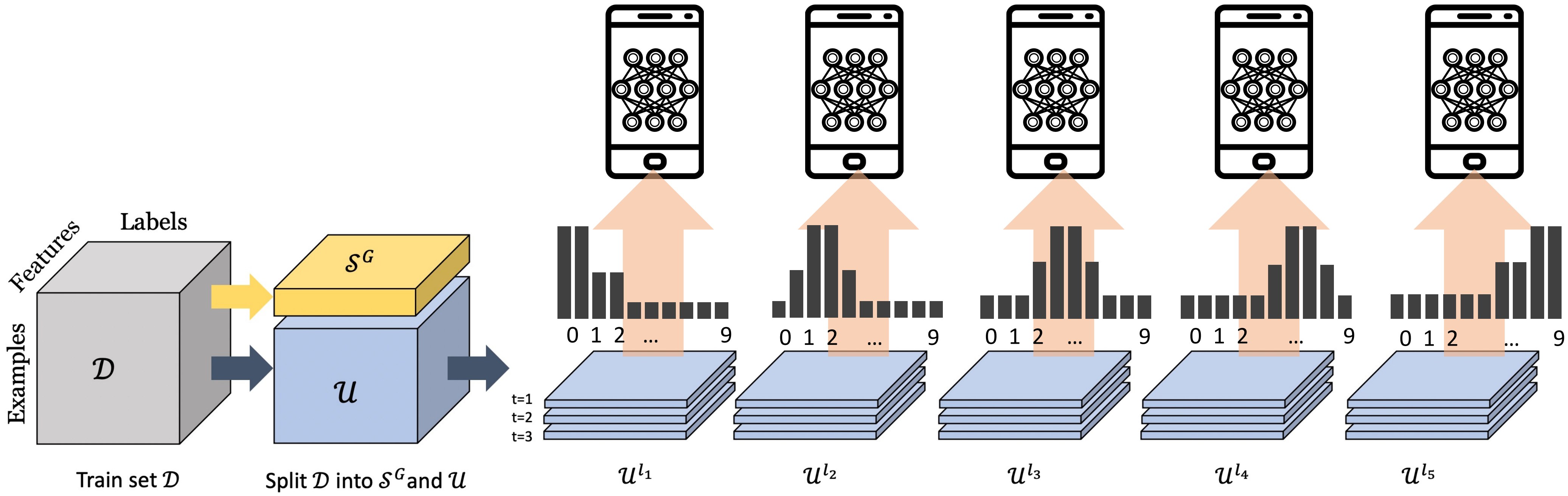
$$minimize\ \mathcal{L}_s(\sigma) = \lambda_s \mathrm{CE}(\mathbf{y}, \mathrm{p}_{\sigma+\psi^*}(\mathbf{y}|\mathbf{x}))$$

For learning on unlabeled data, we minimize **inter-client consistency loss** and **sparsity regularization term** on ψ, as well as **a regularizer** which prevents ψ drifting away from σ^*.

$$minimize\ \mathcal{L}_u(\psi) = \lambda_u \Phi_{\sigma^*+\psi}(\cdot) + \lambda_{L_1} ||\psi||_1 + \lambda_{L_2} ||\sigma^* - \psi||_2^2$$

Experimental Setup

We validate our method on **Batch-IID and -NonIID** datasets (CIFAR-10), which is conventional settings in standard federated learning.



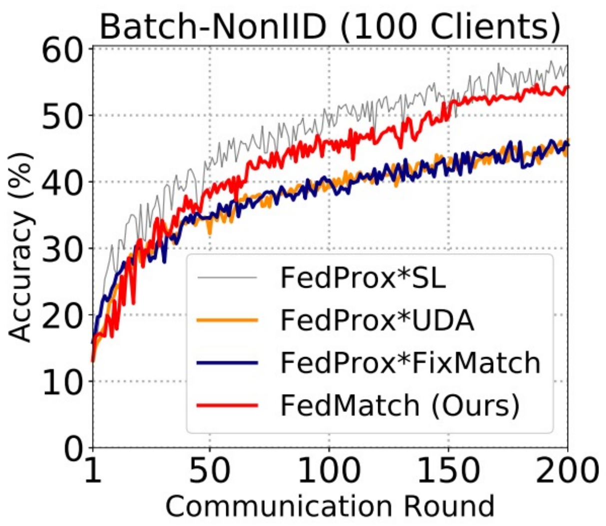
Example Streaming Non-IID dataset under Labels-at-Server scenario

We also evaluate our method on **Streaming Non-IID** dataset (Fashion-MNIST) where locally-generated private data continuously streams in.

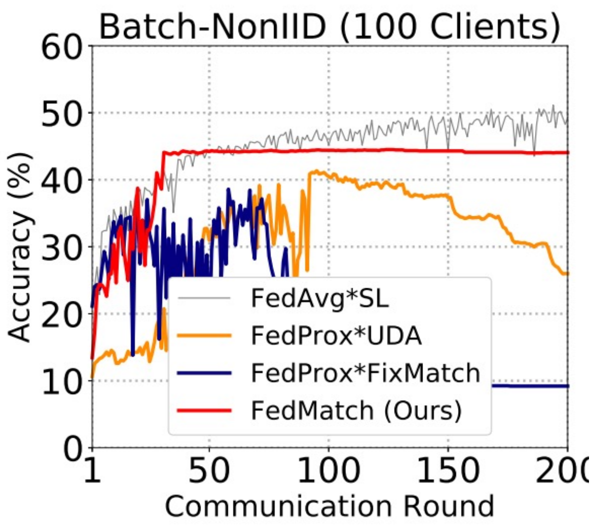
Results of Main Experiments

Our method **outperforms** the naïve Federated Semi-supervised Learning models on both batch and streaming datasets in all scenarios.

Batch-NonIID Dataset with 100 Clients



Labels-at-Client Scenario



Labels-at-Server Scenario

Streaming Non-IID under Labels-at-Client Scenario

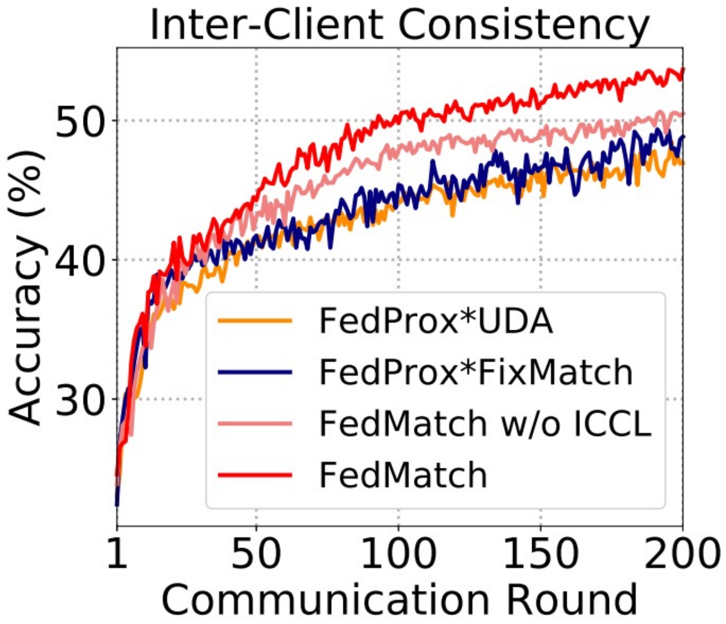
Models	Accuracy	C2S Cost
Local-FixMatch	62.62 ± 0.32 %	N/A
FedProx-FixMatch	62.40 ± 0.43 %	100 %
FedMatch (Ours)	**77.95 ± 0.14 %**	**48 %**

Streaming Non-IID under Labels-at-Server Scenario

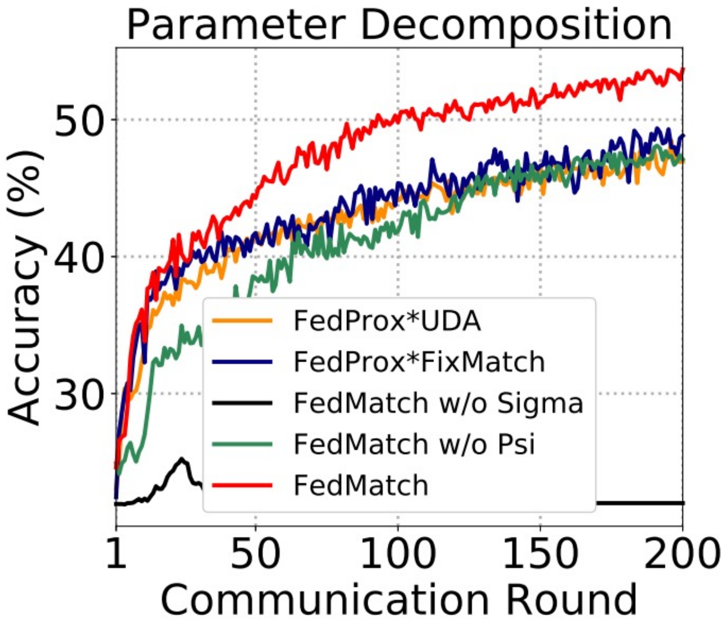
Models	Accuracy	C2S Cost
FedProx-SL	77.43 ± 0.42 %	100 %
FedProx-FixMatch	73.71 ± 0.32 %	100 %
FedMatch (Ours)	**84.15 ± 0.31 %**	**63 %**

Ablation Study

We observe that our inter-client consistency loss plays a significant role to improve performance in FSSL scenario.



Effect of Inter-Client Consistency Loss



Effect of Parameter Decomposition

Our decomposition method also allows us to preserve reliable knowledge from the labeled data.

Conclusion

- We introduce **Federated Semi-Supervised Learning** (FSSL), where each client learns with only partly labeled data (**labels-at-client**), or work with completely unlabeled data with supervised labels only available at the server (**labels-at-server**).
- We propose **FedMatch** which utilizes the **Inter-Client Consistency Loss** to maximize the agreement from different clients, and **Parameter Decomposition** for disjoint learning enabling our model to preserve the reliable knowledge from labeled data.
- We validate our method under Streaming Non-IID and Batch IID settings, which our methods **significantly outperform** naïve approach of FSSL.

Thank You!