FedTGP: Trainable Global Prototypes with Adaptive-Margin-Enhanced Contrastive Learning for Data and Model Heterogeneity in Federated Learning

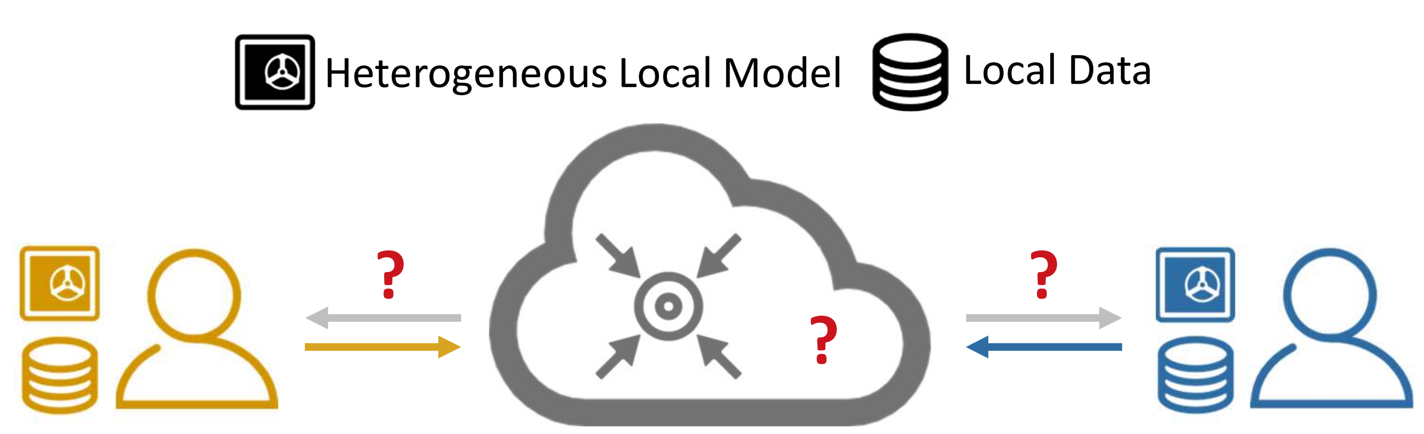
Jianqing Zhang, Yang Liu, Yang Hua, Jian Cao

Background of HtFL

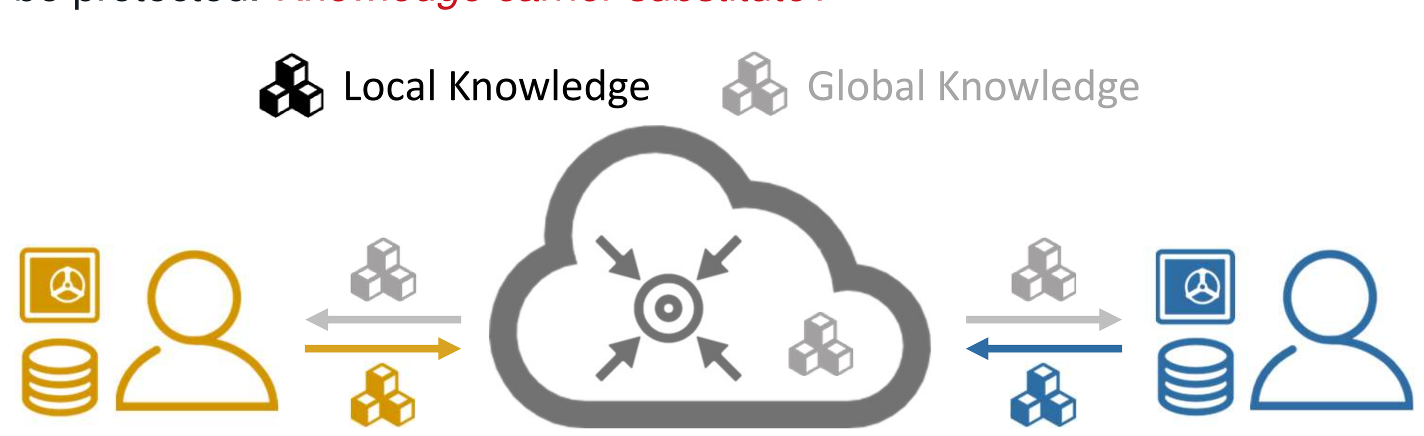
Data heterogeneity: Client-specific private data brings the **data heterogeneity** issue (as shown by the colorful icons below) in Federated Learning (FL). The global model cannot generalize to clients' local data.

Communication Overhead: In the era of large models, typical FL suffers huge communication overhead, as it transmits model parameters. *Lightweight substitute?*

Resource Diversity: Clients using different devices suffer from resource diversity when training homogeneous (same architectures) local models. *Heterogeneous client models?*

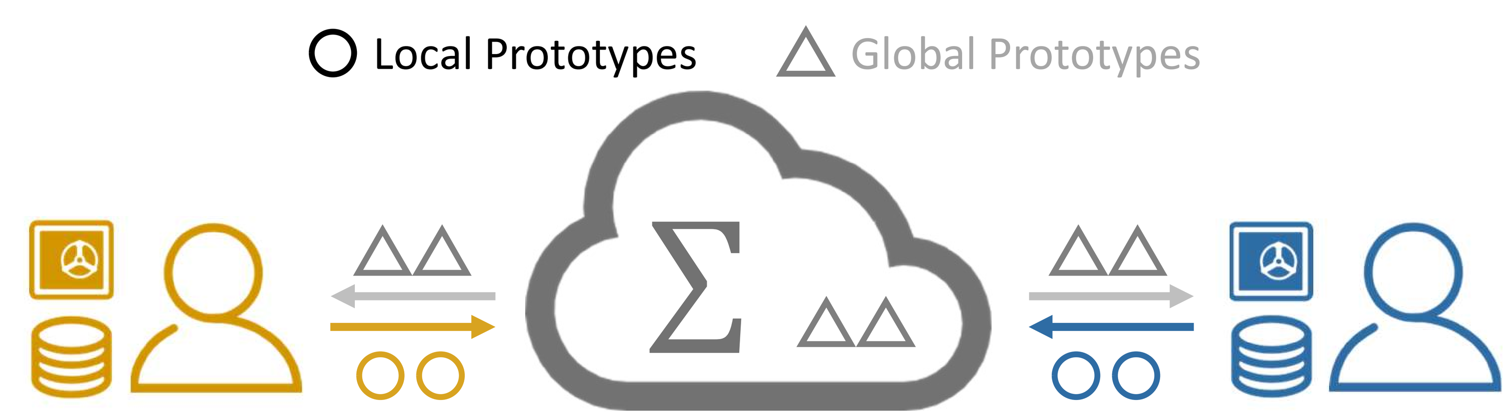


Intellectual Property (IP) Protection: Client model parameters are unique and require substantial effort to obtain, representing a form of IP that should be protected. *Knowledge carrier substitute?*

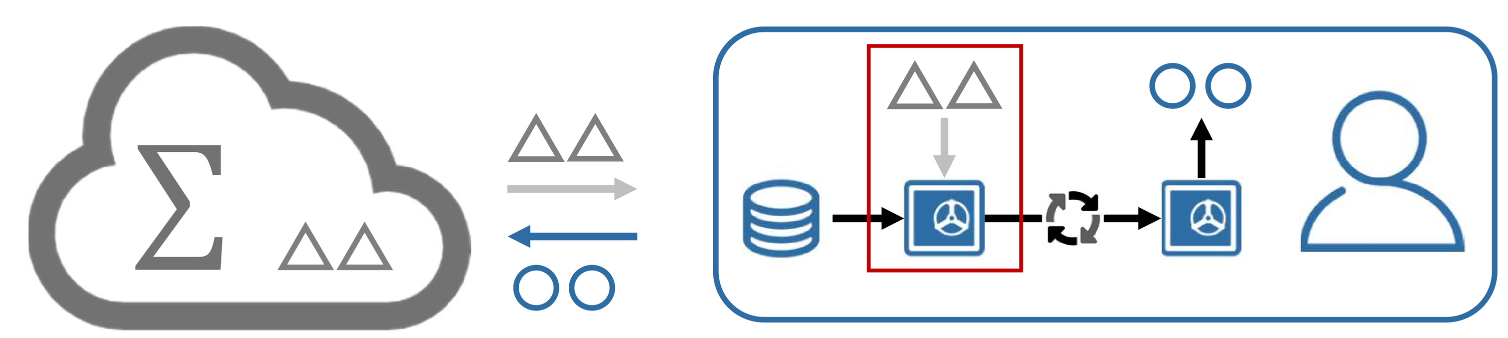


Heterogeneous Federated Learning (HtFL): Considering both data and model heterogeneity, it transmits *lightweight knowledge carriers* instead of exposing model parameters.

Typical HtFL

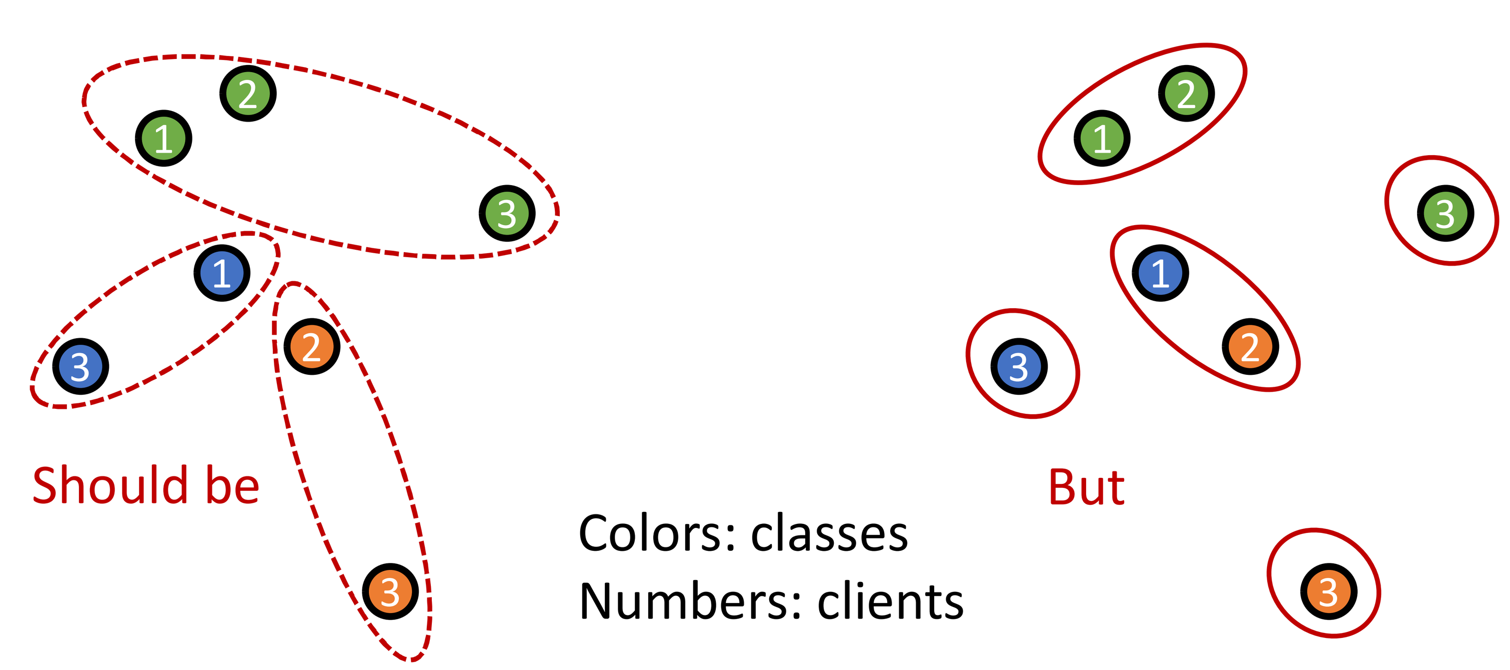


FedProto is a typical method that is *applicable* in HtFL. It transmits *lightweight prototypes* (*i.e.*, class representatives) instead of model parameters and obtains global prototypes through *weighted averaging*.

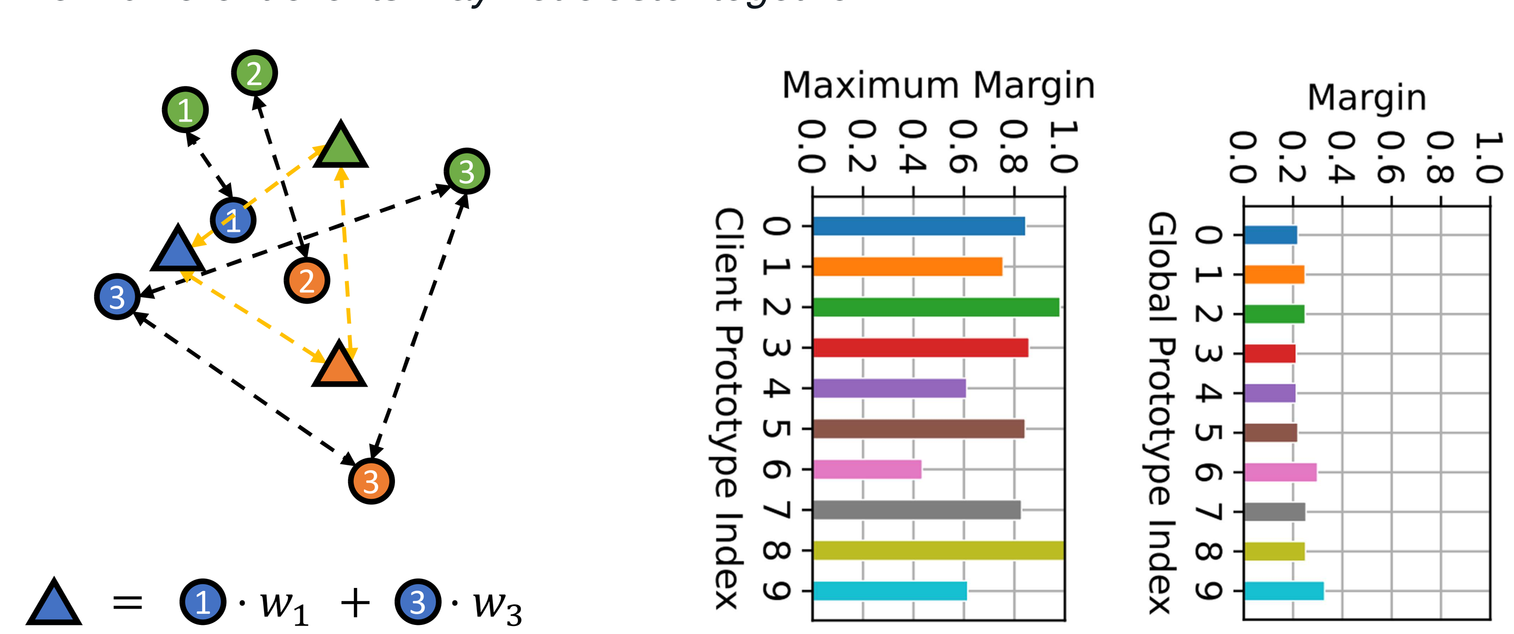


FedProto then guides local training with global prototypes.

Issues of FedProto

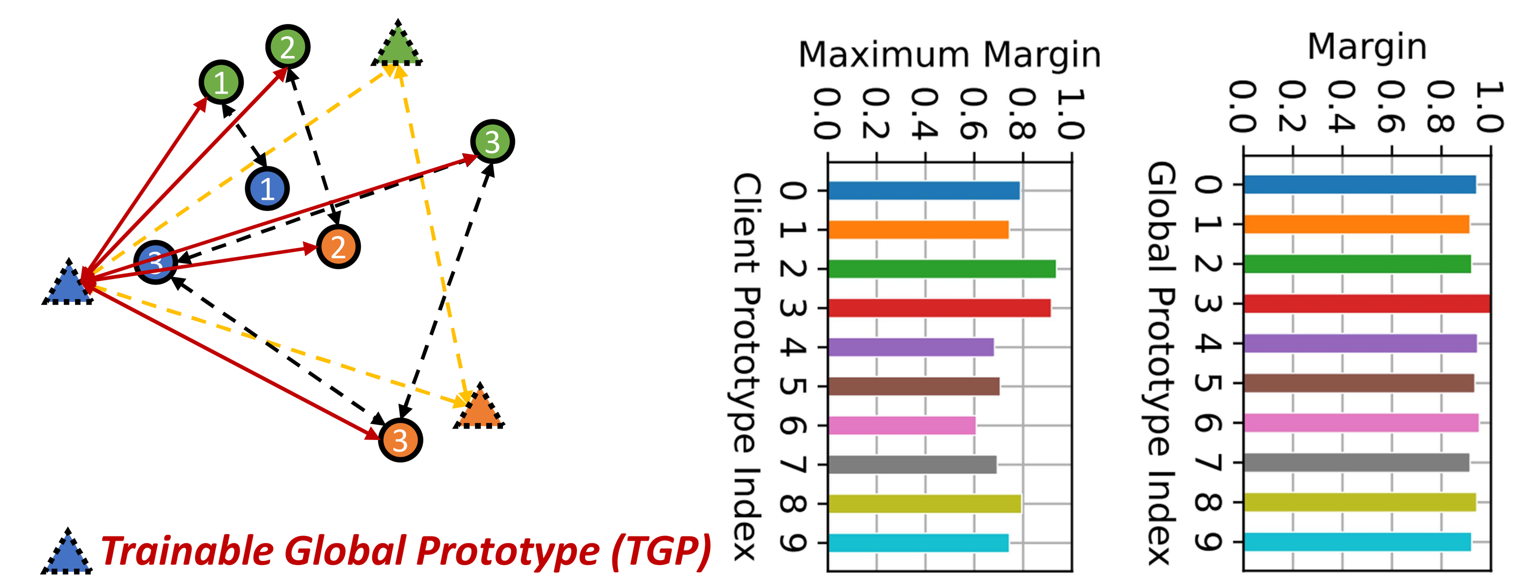


However, weighted averaging is not suitable in HtFL as prototypes are generated on heterogeneous datasets and models, and *prototypes of the same class but from different clients may not cluster together.*



Notice that there are two classes in the central cluster. Global prototype margin *shrinks* after weighted averaging.

Our FedTGP



We remove weighted averaging, consider the uploaded client prototypes as data, and train global prototypes to automatically *enlarge* the global prototype margin. In this way, we can *maintain the discriminative ability of global prototypes*.

We train global prototypes using our proposed ***Adaptive-margin-enhanced Contrastive Learning (ACL)***:

$$\min_{\hat{\mathcal{P}}} \sum_{c=1}^{C} \mathcal{L}_P^c, \tag{1}$$

$$\mathcal{L}_P^c = \sum_{i \in \mathcal{I}^t} -\log \frac{e^{-(\phi(P_i^c, \hat{P}^c)+\delta(t))}}{e^{-(\phi(P_i^c, \hat{P}^c)+\delta(t))} + \sum_{c'} e^{-\phi(P_i^c, \hat{P}^{c'})}}, \tag{2}$$

$$\delta(t) = \min(\max_{c \in [C], c' \in [C], c \neq c'} \phi(Q_t^c, Q_t^{c'}), \tau), \tag{3}$$

where $\hat{\mathcal{P}} = \{\hat{P}^c\}_{c=1}^{C}$, ϕ measures the Euclidean distance, $c' \in [C], c' \neq c$, $\mathcal{I}^t$ is the participating client set at tth iteration with client participation ratio ρ, τ is a margin threshold, and $Q_t^c = \frac{1}{|\mathcal{P}_t^c|} \sum_{i \in \mathcal{I}^t} P_i^c, \forall c \in [C]$.

Experiments

We evaluate our FedTGP on scenarios with 12 widely-used model architectures on $20 \sim 100$ clients. Our FedTGP surpasses counterparts by up to **9.08%** and also surpasses FedProto by up to **13.85%**. "C": Cifar, "F102": Flowers102, "TINY": Tiny-ImageNet.

Settings	Pathological Setting				Practical Setting			
Datasets	C10	C100	F102	TINY	C10	C100	F102	TINY
LG-FedAvg	86.82	57.01	58.88	32.04	84.55	40.65	45.93	24.06
FedGen	82.83	58.26	59.90	29.80	82.55	38.73	45.30	19.60
FML	87.06	55.15	57.79	31.38	85.88	39.86	46.08	24.25
FedKD	87.32	56.56	54.82	32.64	86.45	40.56	48.52	25.51
FedDistill	87.24	56.99	58.51	31.49	86.01	41.54	49.13	24.87
FedProto	83.39	53.59	55.13	29.28	82.07	36.34	41.21	19.01
FedTGP	**90.02**	**61.86**	**68.98**	**34.56**	**88.15**	**46.94**	**53.68**	**27.37**

- **Full paper:** https://arxiv.org/abs/2401.03230
- **Code:** https://github.com/TsingZ0/FedTGP
- **HtFL Algorithm Library:** https://github.com/TsingZ0/HtFLlib