NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Personalized Continual Federated Learning

Tan Ziying Alysa

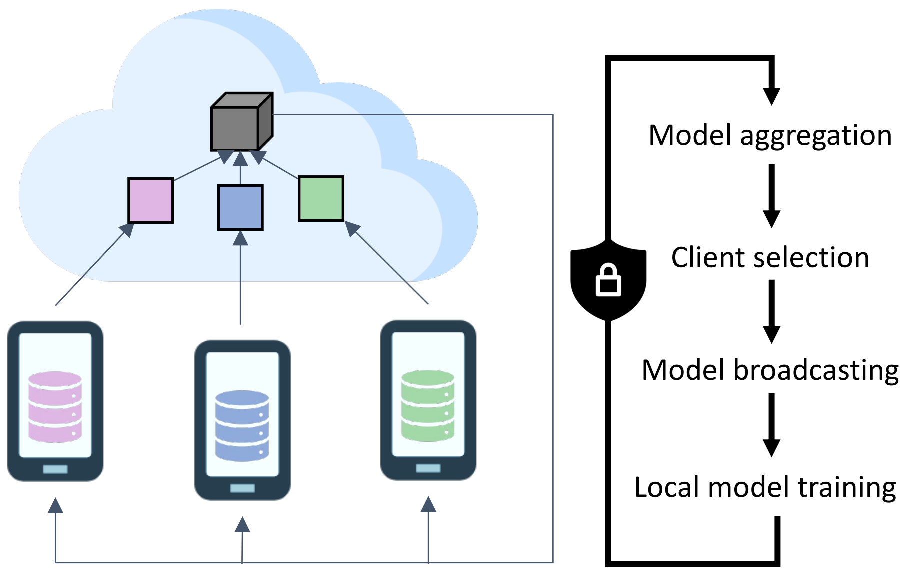
PhD Student, Alibaba-NTU Joint Research Institute

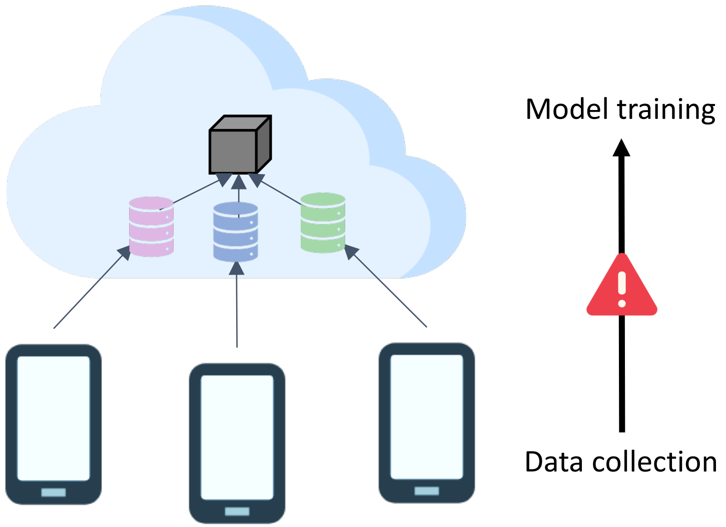
Agenda

- Background and Key Concepts
  - Federated Learning (FL)
  - Personalized Federated Learning (PFL)
  - Continual Learning (CL)
- Personalized Continual Federated Learning (PCFL)
  - Goal and open research questions

Federated Learning (FL)

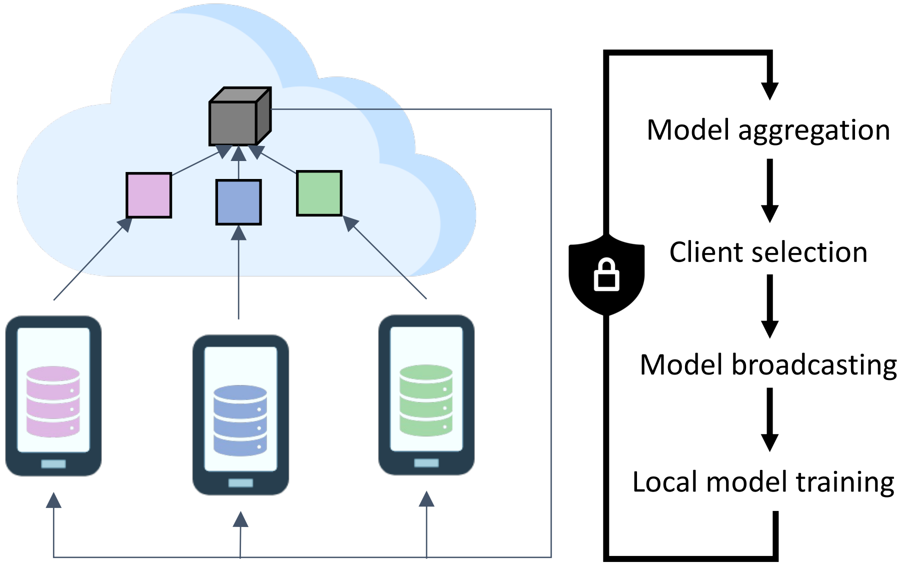
Goal of FL: to collaboratively train a ML model on distributed data





Federated Learning	Centralized Machine Learning
✔ Generalization ✔ Privacy ✔ Communication	✔ Generalization

Federated Learning (FL)



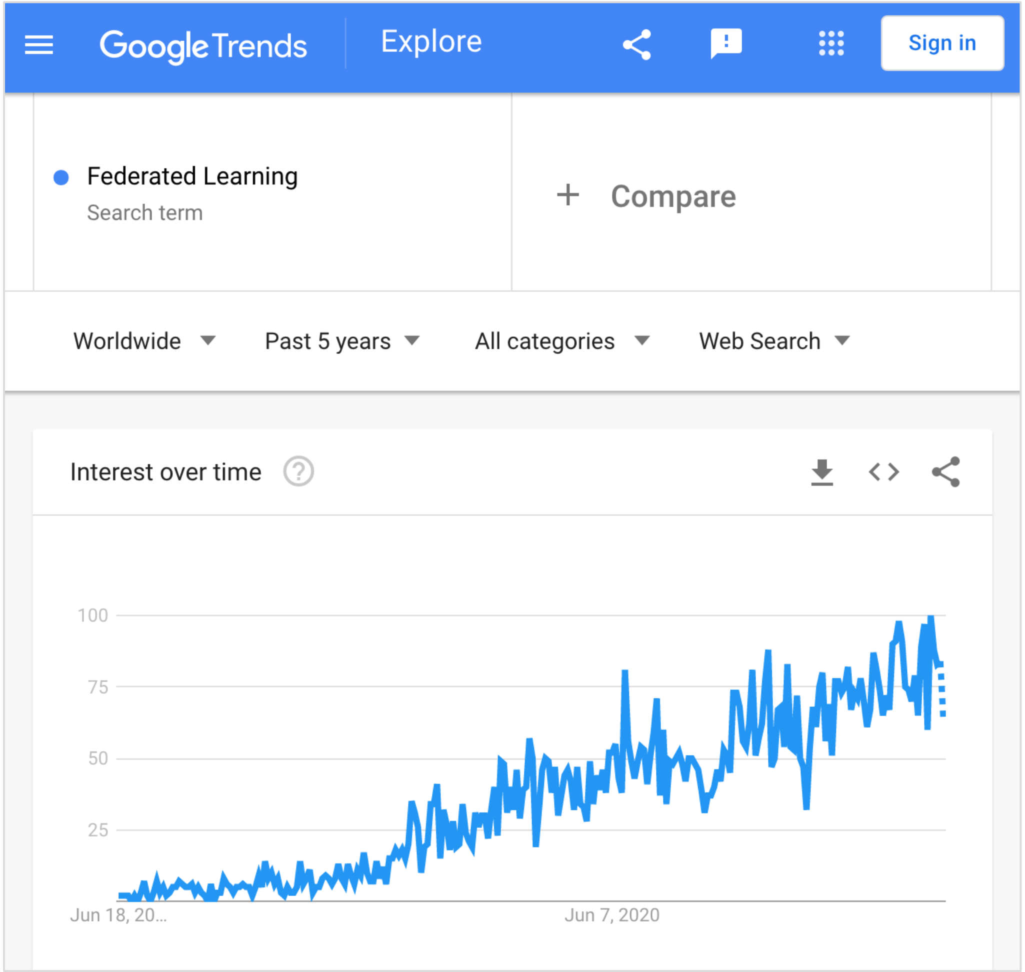
Algorithm 1 Federated Averaging

Inputs: Number of communication rounds T; Number of local epochs E; Size of minibatch B; Learning rate η
Outputs: Aggregated server parameters $\boldsymbol{\theta}_t$

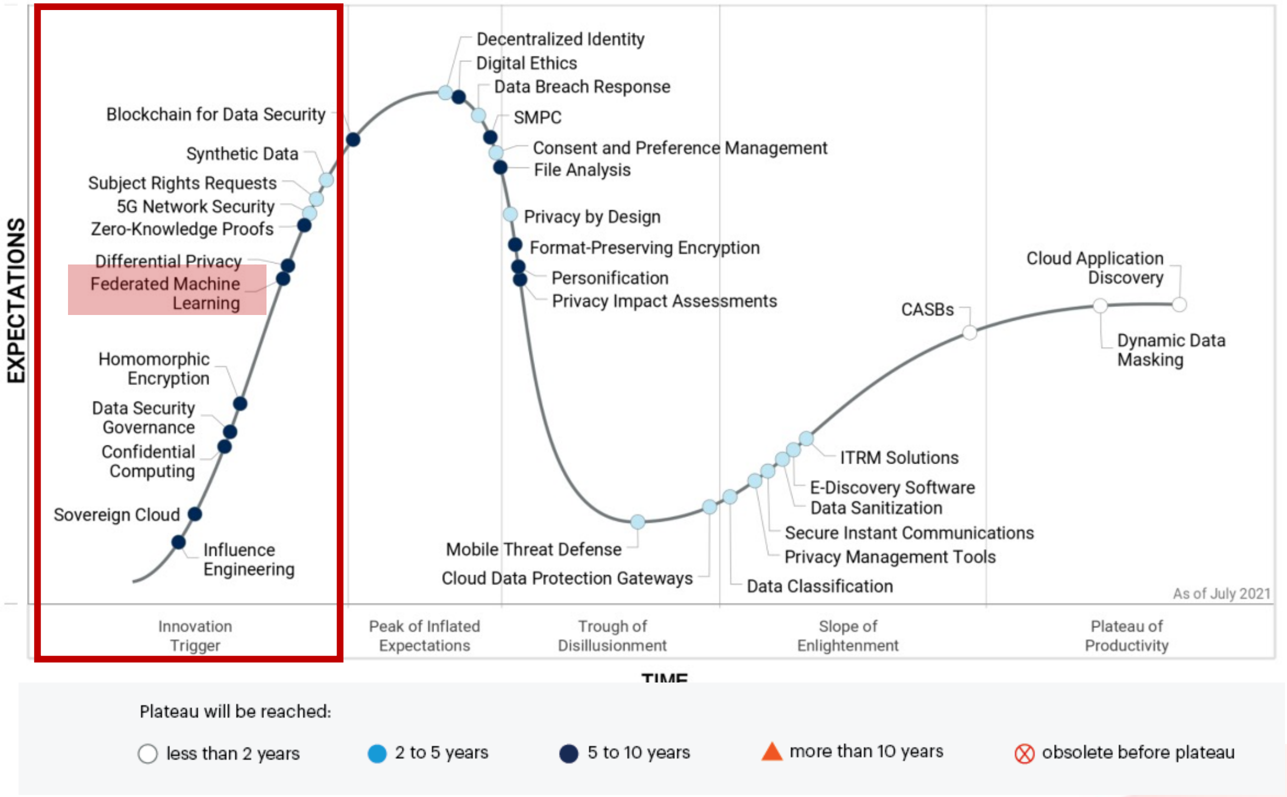
```
procedure FEDAVG
    ServerUpdate:
        Initialize parameters θ_0
        for round t ∈ {1, 2, ··· , T} do
            C_t ← (random subset of clients)
            for client c ∈ C_t do
                θ^c_{t+1} ← ClientUpdate(c, θ_t)
            end for
            θ_{t+1} ← Σ_c (n_c / n) θ^c_{t+1}
        end for

    ClientUpdate(c, θ_t):
        B ← (split local data into batches of size B)
        for local epoch e ∈ {1, 2, ··· , E} do
            for batch b ∈ B do
                θ ← θ − η∇ℓ(θ; b)
            end for
        end for
        return local parameters θ
end procedure
```

[McMahan et al., 2017]

Why FL?



[Google Trends, 2022]



[Gartner Hype Cycle for Privacy, 2021]

NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Performance Issues with Vanilla FL

I. Poor convergence on non-IID data

- Client drift occurs when the local distributions are highly different from the global distribution
- Server updates move towards the average of client optima $\frac{x_1^*+x_2^*}{2}$ instead of the true global optimum x^*

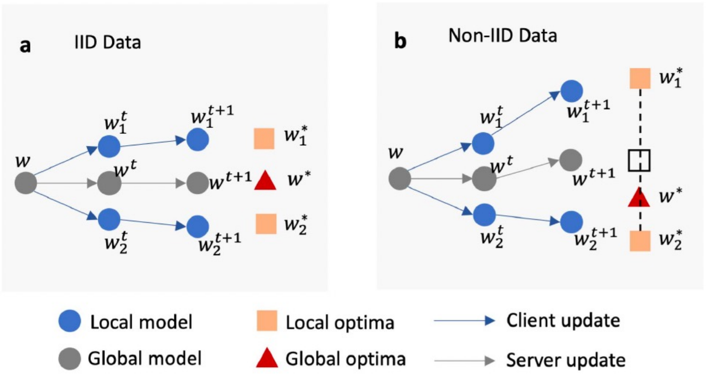
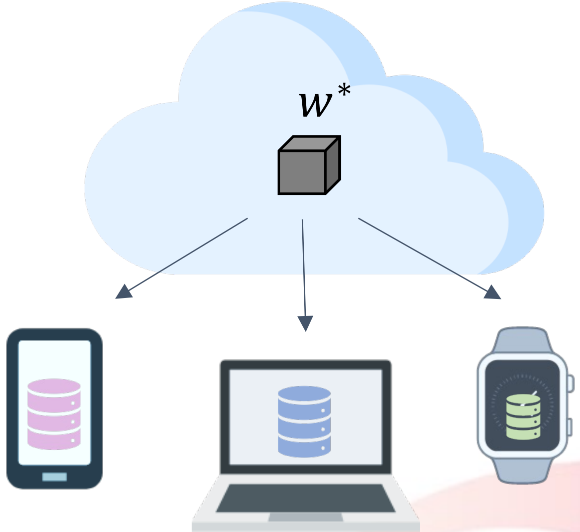


Fig. 2. Illustration of client drift in FedAvg for two clients with two local steps. (a) IID data setting. (b) Non-IID data setting.

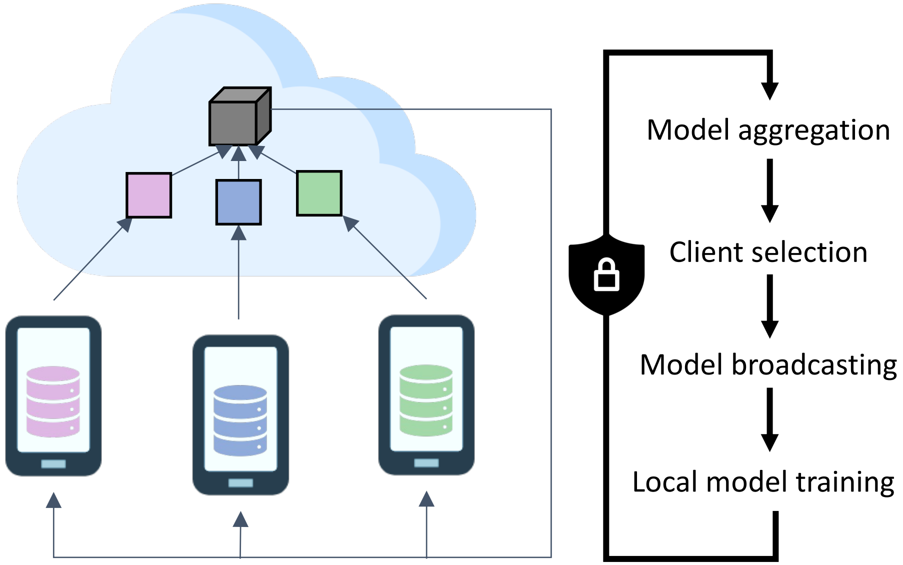
II. Lack of solution personalization

- Trains and makes inference a single globally-shared FL model
- Designed to fit the "average client"
- The global model does not generalize well for data distributions that are different from the global distribution



NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Towards Personalized Federated Learning (PFL)



Federated Learning

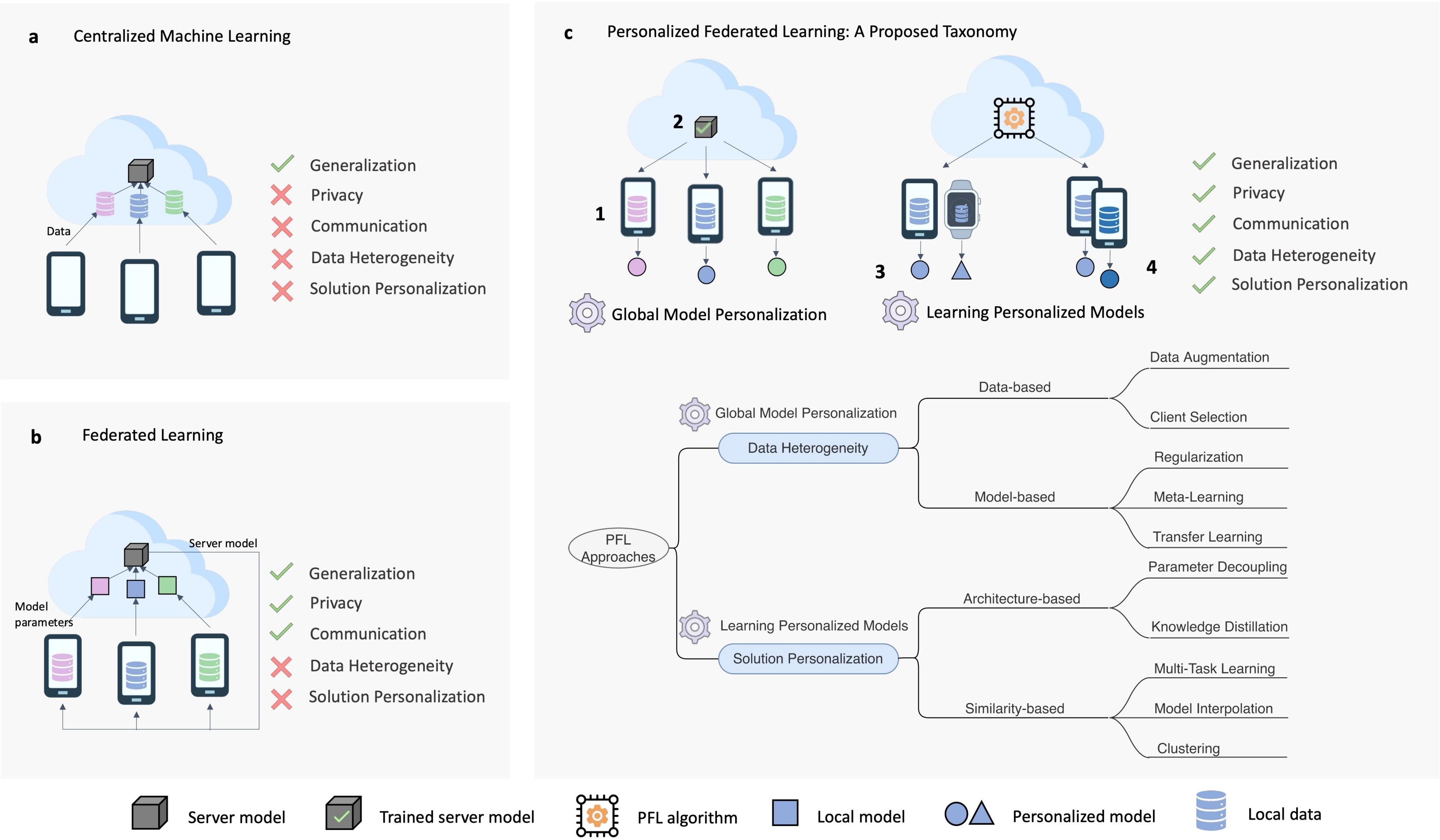
✓Generalization ✓Privacy ✓Communication

Personalized Federated Learning

✓Generalization ✓Privacy ✓Communication

✓Heterogeneity ✓Personalization

Proposed PFL Taxonomy



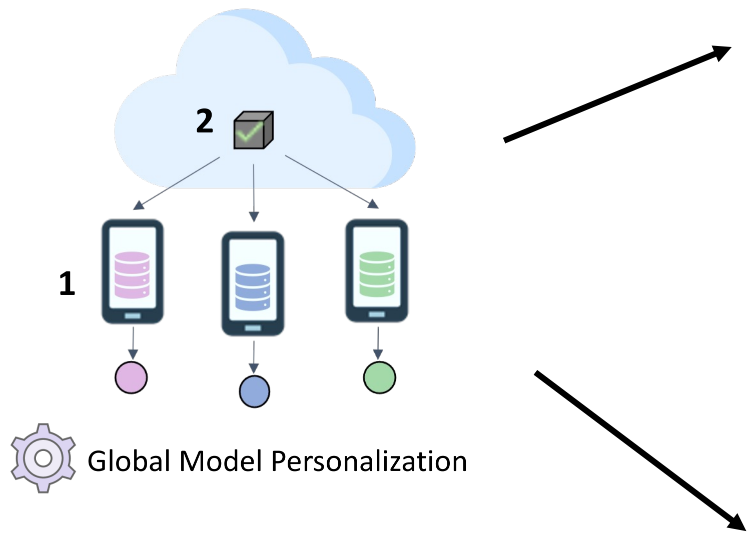
***Alysa Ziying Tan**, Han Yu, Lizhen Cui, and Qiang Yang, "Towards personalized federated learning," IEEE Transactions on Neural Networks and Learning Systems, 2022.*

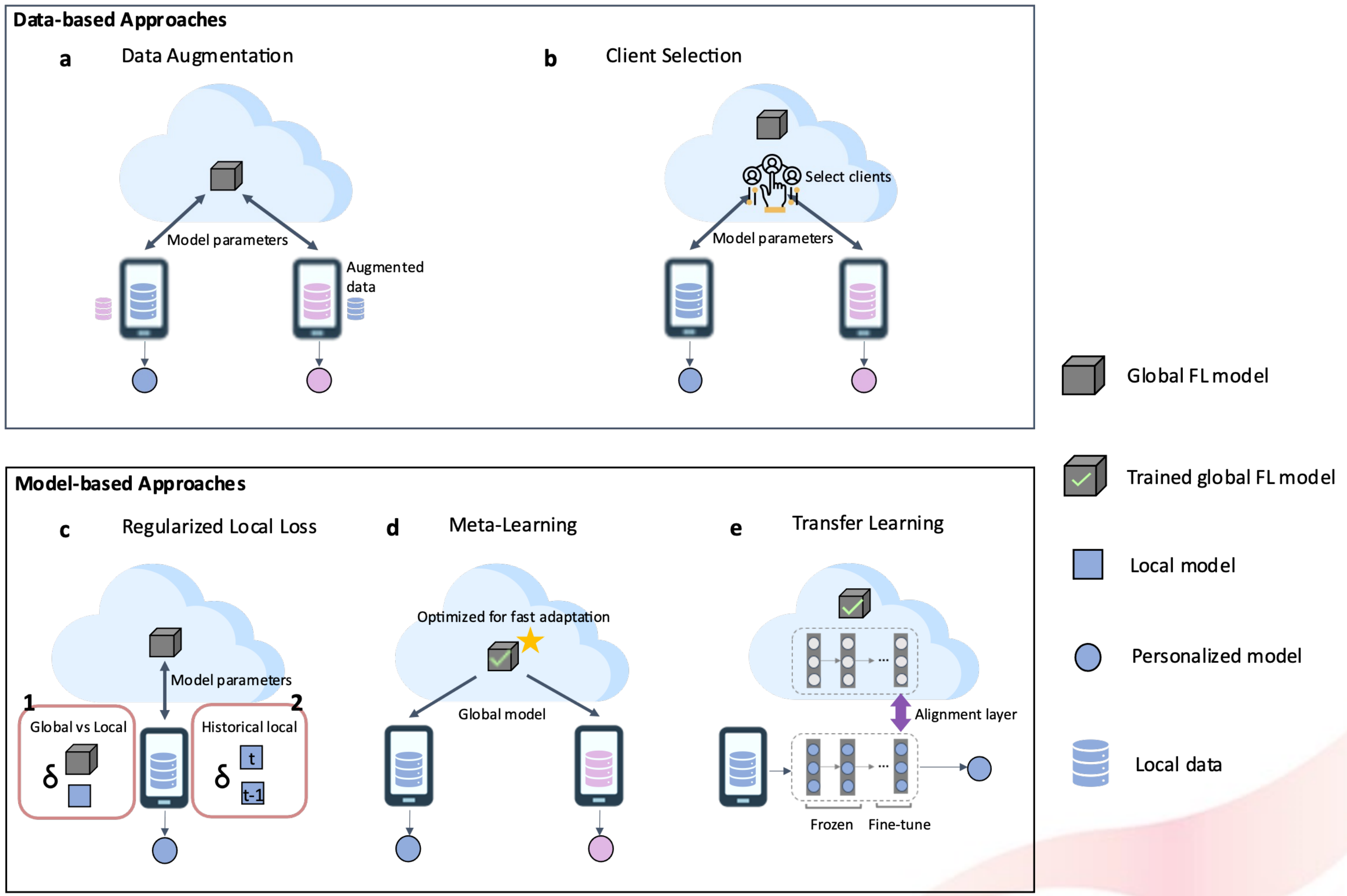
NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Strategy 1: Global Model Personalization

Goal of PFL: to improve the performance of the global FL model under data heterogeneity

"single global model setting"





Data–based Approaches

Reduces the heterogeneity of data distributions

(i) Data Sharing

- **[Zhao et al., 2018]**
  - Distributes a small amount of global proxy data (uniform distribution over classes) to the clients

(ii) Data Augmentation

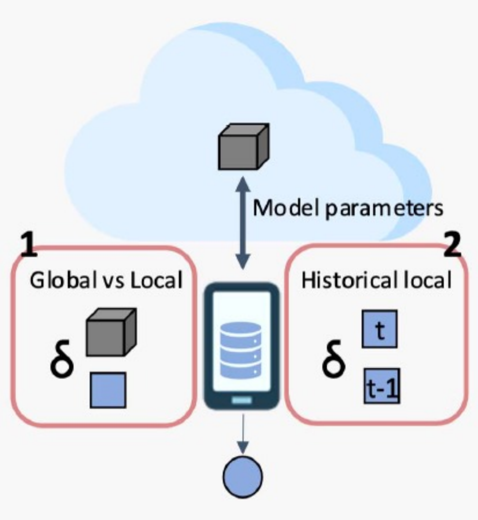
- **FAug [Jeong et al., 2018]**
  - Data samples of minority classes are uploaded to the server to train the GAN model in the server
  - The GAN model is sent to clients to augment its local data towards yielding an IID dataset
- **Astraea [Duan et al., 2021]**
  - Uses Z-score based augmentation & down-sampling to reduce class imbalance

(iii) Client Selection

- **FAVOR [Wang et al., 2020]**
  - Proposed a deep Q-learning formulation to mitigate the bias introduced by non-IID data
  - Selects a subset of clients in each training round that maximizes the reward in terms of accuracy and penalizes the use of more communication rounds

Regularization

Limits the impact of local updates to achieve convergence stability & improve the generalization of the global model



(i) Between global & local models

- **FedProx [Li et al., 2020]**

$$\frac{\mu}{2}\|\theta_c - w\|^2$$

L2-norm

- **FedCL [Yao & Sun, 2020]**

$$\mu \sum_{i,j} \boxed{\Omega_{ij}} (\theta_{c,ij} - w_{ij})^2$$

Importance matrix estimated on proxy data in server

Elastic Weight Consolidation

- **Scaffold [Karimireddy et al., 2020]**

$$v - v_c$$

Estimated difference of update directions between global & local models

Variance reduction

(ii) Between historical local model snapshots

- **MOON [Li et al., 2021]**

$$-\mu \log \frac{exp(sim(\theta_c, w)/T)}{exp(sim(\theta_c, w)/T) + exp(sim(\theta_c, \theta_c^{t-1})/T)}$$

Contrastive learning

- Reduce distance between global & local models to reduce client drift
- Increase distance between local model snapshots to speed up convergence

NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Meta-Learning

Learns a global model initialization for fast adaptation on a new heterogeneous task ("client")

Standard FL

$$\min_{w \in \mathbb{R}^d} F(w) := \frac{1}{C} \sum_{c=1}^{C} f_c(w)$$

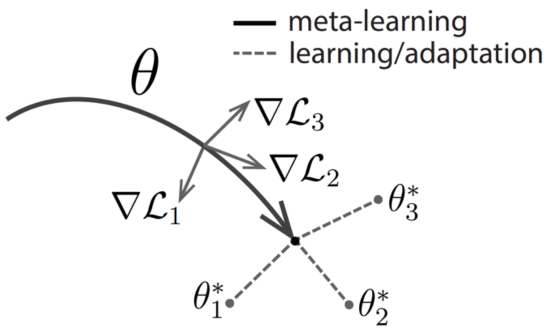
Per-FedAvg [Fallah et al., 2020]

- Proposed a variant of FedAvg that builds on the MAML [Finn et al., 2017] formulation
- Goal is to learn a global model that performs well on a new task after it is updated with a few steps of gradient descent

$$\min_{w \in \mathbb{R}^d} F(w) := \frac{1}{C} \sum_{c=1}^{C} \boxed{f_c(w - \alpha \nabla f_c(w))} \rightarrow F_c(w)$$

Meta-function associated with client c

Min average of meta-functions



- Gradient computation requires access to second-order information -> computationally expensive

$$\nabla F_c(w) = (I - \alpha \nabla^2 f_c(w)) \nabla f_c(w - \alpha \nabla f_c(w))$$

- Use of gradient approximations e.g. FO-MAML [Finn et al., 2017] , HF-MAML [Fallah et al., 2020]

Transfer Learning

Reduces the domain discrepancy between the trained global FL model and the local model

FedHealth [Chen et al., 2020]

- Introduces an alignment layer to adapt the second-order statistics of the source & target domains

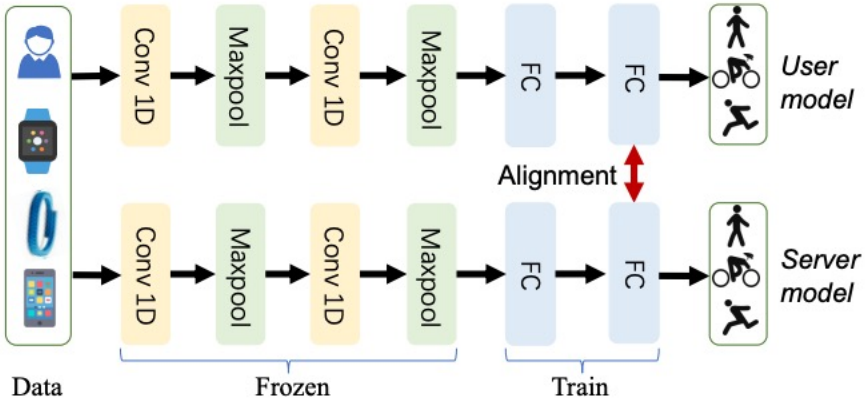
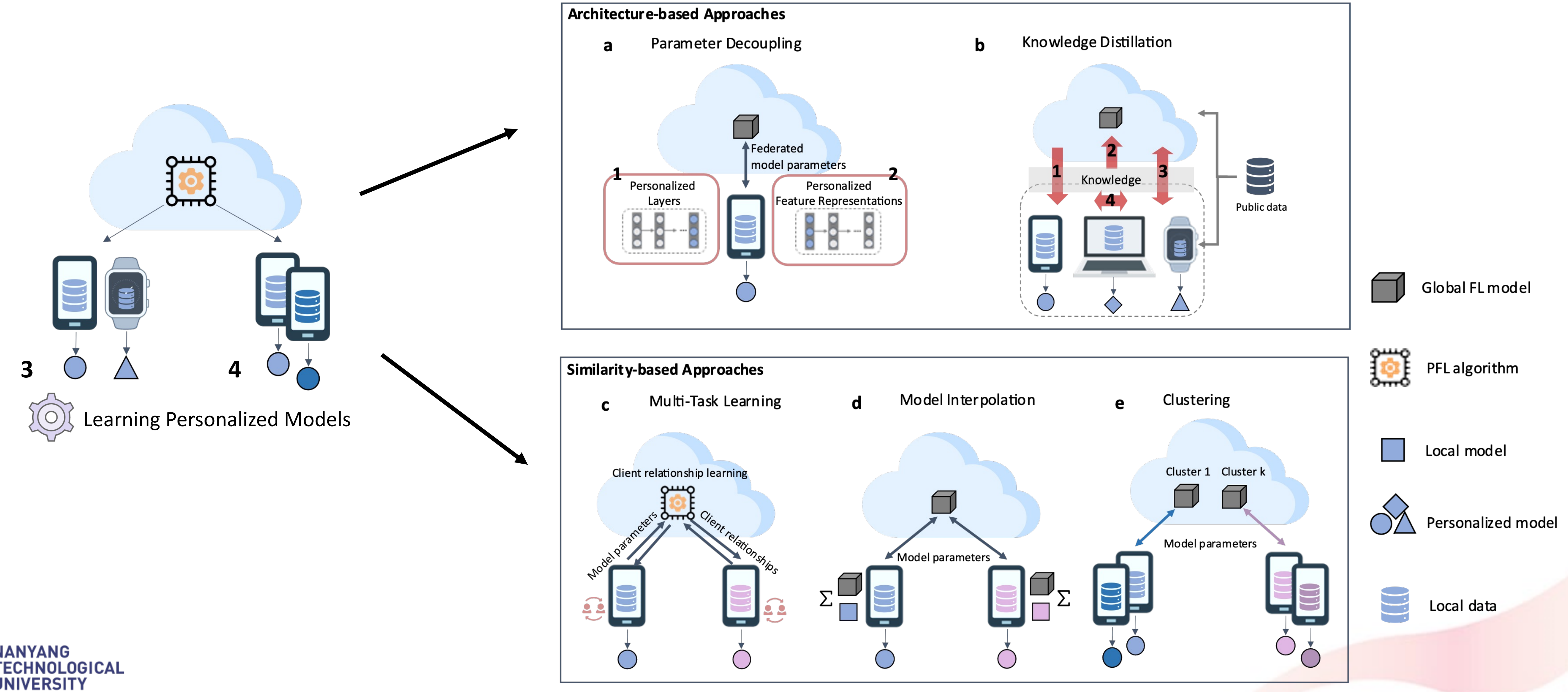


Figure 3: The transfer learning process of FedHealth

Strategy 2: Learning Personalized Models

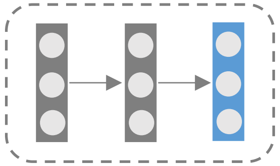
Goal of PFL: to collaboratively train individual personalized models for each client



Parameter Decoupling

Comprises private and federated parameters

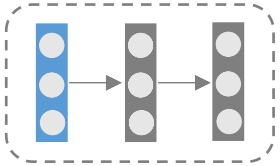
(i) Personalized layers



[Arivazhagan et al., 2019]

- Personalized layers are kept private at the clients for local training, base layers are used in FL

(ii) Personalized feature representations



FURL [Bui et al., 2019]

- User embeddings as private parameters; character embeddings, LSTM and MLP layers as federated parameters.

LG-FedAvg [Liang et al., 2020]

- Combines local representation learning and global federated training
- Specialized encoders can be designed based on the source data modality (e.g. image, text)
- Fair and unbiased representations may be learnt

(iii) Learning the privatization strategy **[Li et al., 2021]**

NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Knowledge Distillation

Allows a personalized architecture design for each client

FedMD [Li & Wang, 2019]

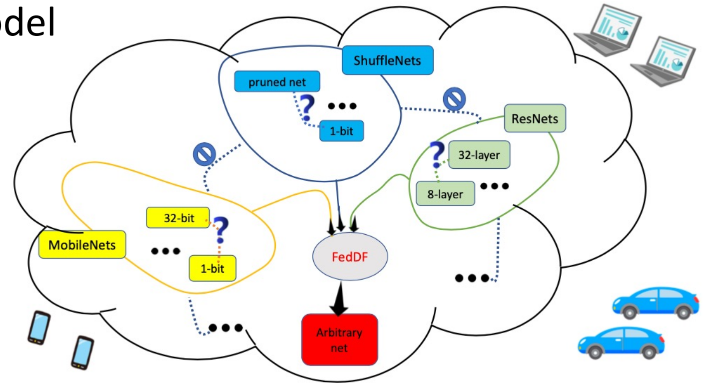
- Each client designs its own personalized model
- Learns through a consensus result using the average class scores on a public dataset.
- For every communication round, each client trains its model on the public dataset to approach the updated consensus, and fine-tunes its model on its private dataset thereafter.

FedDF [Lin et al., 2021]

- The server constructs p prototype models to represent clients with identical model architectures (e.g. ResNet, MobileNet).
  - Step 1: Perform FedAvg within each prototype group to initialize student model
  - Step 2: Perform ensemble distillation for cross-architecture learning

$$\min_{w_p \in \mathbb{R}^d} F(w) := \mathbb{E}_{x \sim D_p}\left[KL\left[\sigma\left(\frac{1}{C}\sum_{c=1}^{C} g(\theta_c; x)\right), \sigma\left(g(w_p; x)\right)\right]\right]$$

Client teacher model Prototype model



Multi-Task Learning

Learns personalized models while leveraging task (“client”) relationships

MOCHA [Smith et al., 2017]

$$\min_{\mathbf{W},\mathbf{\Omega}} \left\{ \sum_{c=1}^{C} \sum_{i=1}^{n_i} \ell\left(\mathbf{w}; x_i; y_i\right) + \mu_1 tr(\mathbf{W}\boxed{\mathbf{\Omega}}\mathbf{W}^T) + \mu_2 \|\mathbf{W}\|^2 \right\}$$

Relationship matrix of learning tasks

- Extends MTL to FL
- Learns a personalized model for each client, related clients learn similar models
- Uses a primal-dual formulation, only for convex models

FedAMP [Huang et al., 2021]

- Maintains a personalized cloud model u_c for each client in the server
- Enforces stronger pairwise collaboration for clients with similar data distributions

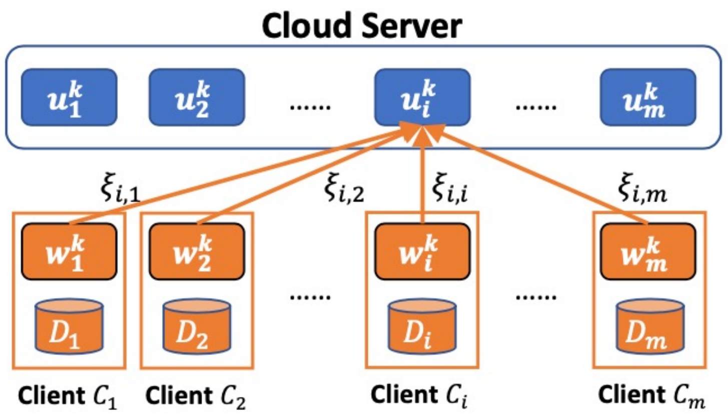
$$u_c = \xi_{c,1} w_1 + \ldots + \xi_{c,m} w_m$$

$$\xi_{i,j} = \alpha_k \boxed{A'\left(\|\mathbf{w_i^{k-1}} - \mathbf{w_j^{k-1}}\|^2\right)}, (i \neq j)$$

Similarity function

- u_c is transferred to each client to perform local training

$$w_c^* = argmin_{w \in \mathbb{R}^d} f_c\left(w\right) + \frac{\mu}{2\alpha}\|w - u_c\|^2$$



Model Interpolation

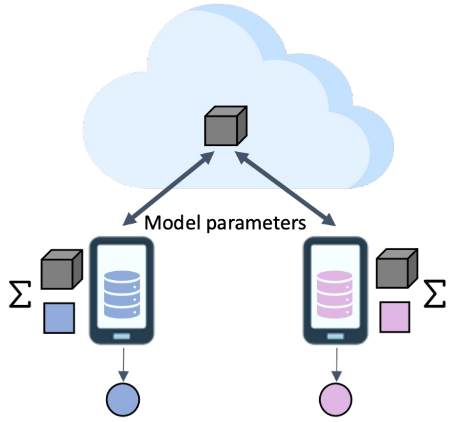
Learns personalized models using a mixture of global and local models

[Hanzely & Richtarik, 2020]

- Each client learns a personalized model θ_c
- The personalized model is encouraged not to depart too far from the mean
  - $\lambda \to 0$, local model learning
  - $\lambda \to \infty$, global model learning

$$\min_{\theta_1,\ldots,\theta_c \in \mathbb{R}^d} F(\theta) := \{\boxed{f(\theta)} + \lambda \boxed{g(\theta)}\}$$

$$\frac{1}{C}\sum_{c=1}^{C} f_c(\theta_c) \qquad g(\theta) := \frac{1}{2C}\sum_{c=1}^{C}\|\theta_c - \bar{\theta}\|^2$$



APFL [Deng et al., 2020]

- Introduces a mixing parameter that is adaptively learnt during the FL training process to control the balance between the global and local models

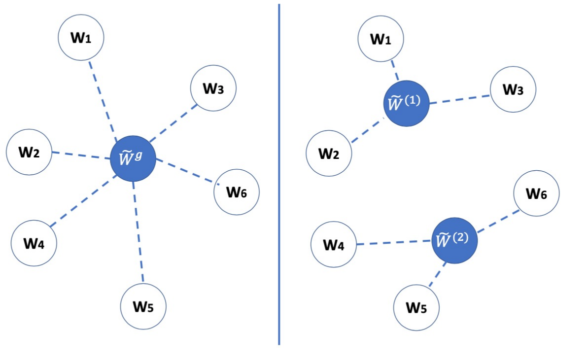
$$\theta_c^* = argmin_{\theta \in \mathbb{R}^d} f_c\left(\boxed{\alpha_c}\theta + (1-\alpha_c)\, w\right)$$

Larger weighting factor if local & global data distributions are not well-aligned

Clustering

Supports group level personalization

- For applications where there are inherent partitions among clients or data distributions that are significantly different
- A multi-model approach where an FL model is trained for each homogeneous client cluster



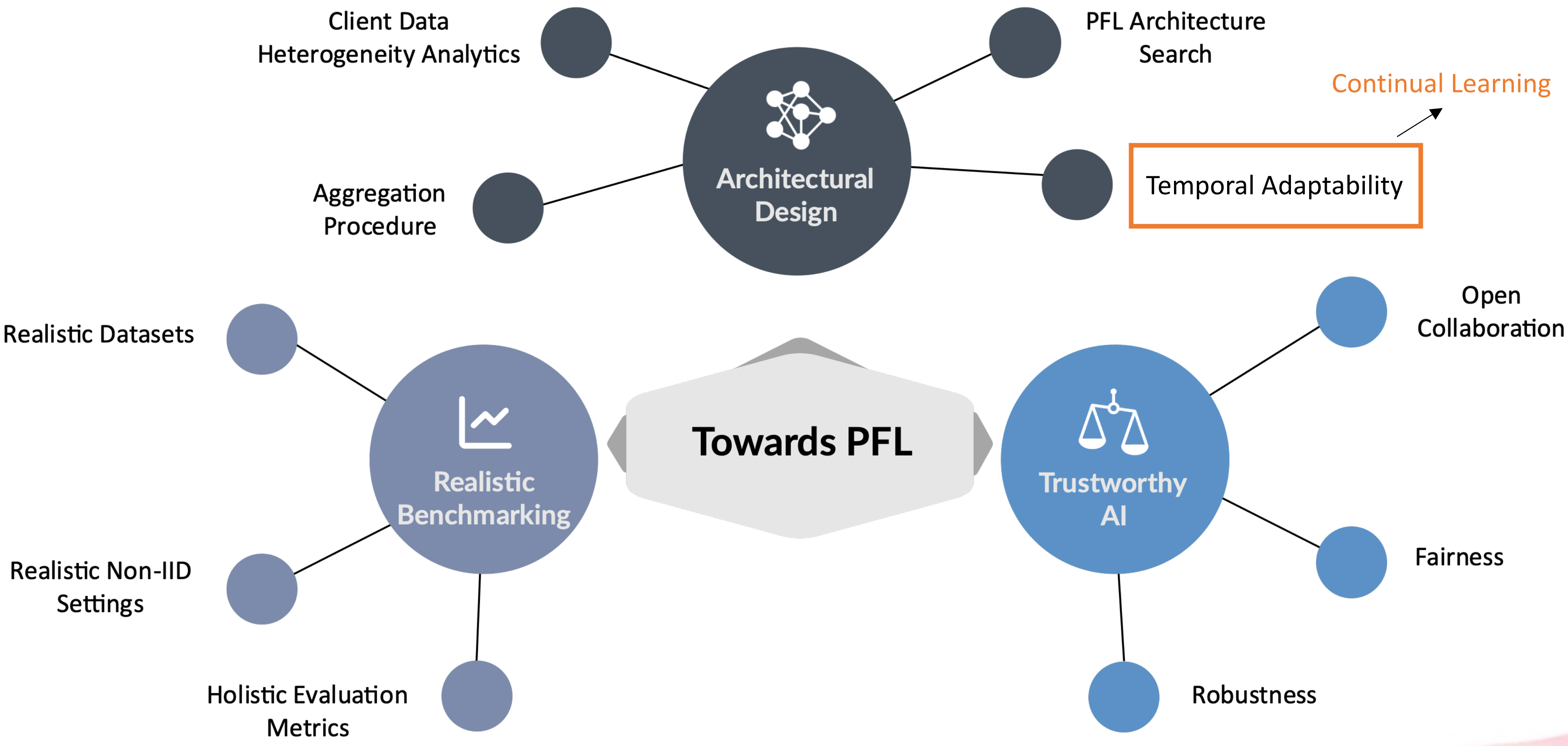
- **FL+HC [Briggs et al., 2020]**
  - Applies agglomerative hierarchical clustering based on global and local model parameter differences
  - FL training is then performed independently for each client cluster to produce c federated models

- **CBFL [Huang et al., 2019]**
  - Applies K-means clustering to cluster clients based on the encoded features of their private data
  - A FL model is then trained for each cluster

- **FeSEM [Xie et al., 2020]**
  - Proposed a multi-center formulation that learns multiple global models
  - Uses expectation maximization to solve a joint optimization problem with distance-based multi-center loss

NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

PFL Research Directions



NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Continual Learning

(aka Incremental learning, Lifelong learning)

Goal of CL: learn new knowledge from a new experience (task) without forgetting knowledge learnt from old experiences (tasks)

- 3 key scenarios studied in CL research

Learning on a sequence of tasks

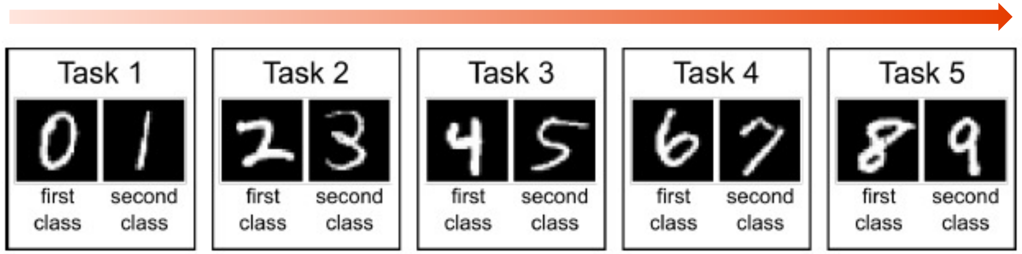


Figure 1: Schematic of split MNIST task protocol.

Table 2: Split MNIST according to each scenario.

Multiple distinct tasks	**Task-IL**	With task given, is it the 1st or 2nd class? (e.g., 0 or 1)
Changing data distributions	**Domain-IL**	With task unknown, is it a 1st or 2nd class? (e.g., in $[0, 2, 4, 6, 8]$ or in $[1, 3, 5, 7, 9]$)
New classes	**Class-IL**	With task unknown, which digit is it? (i.e., choice from 0 to 9)

[Van et al., 2019]

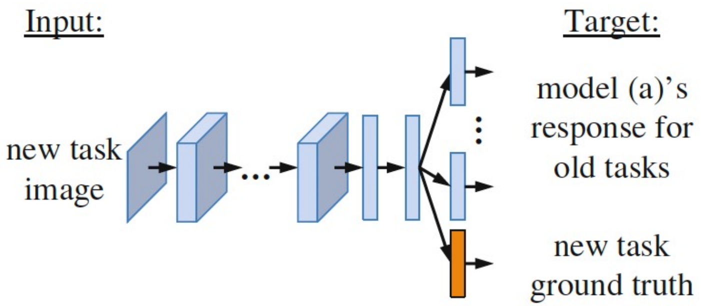
Continual Learning Approaches

1) Replay-based methods

- **Rehearsal**: store samples in raw format, reuse as model inputs for training
  - iCARL [Rebuffi et al., 2017]: nearest-mean-of-exemplars
  - REMIND [Hayes et al., 2020]: quantized convolutional features
  - Requires storage, privacy risks, prone to overfitting
- **Pseudo rehearsal**: generate pseudo-samples/features in-memory to avoid exemplar storage
  - Challenging on complex datasets, relies on the quality of the generated synthetic samples.

2) Regularization-based methods

- Introduce regularization terms in the loss function to constrain weights updates to prevent forgetting
- Knowledge distillation: prevent the deviation of model outputs from a teacher model that has been trained on old classes
  - LwF [Li et al., 2016]



  - Cross-distilled loss [castro et al., 2018] , pooled outputs distillation loss [Douillard et al., 2020], attention distillation loss [Dhar et al., 2019]

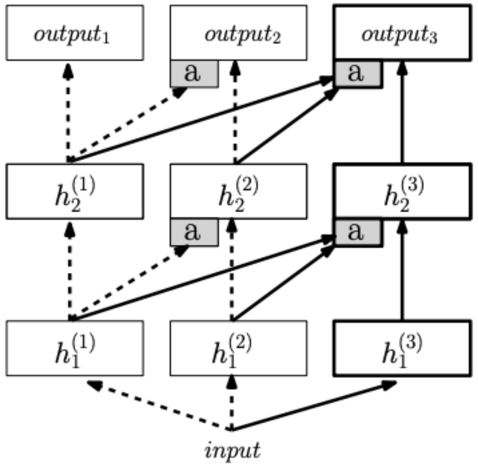
NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Continual Learning Approaches

3) Architecture-based methods

- Dedicates different model parameters to each task to prevent forgetting
  - HAT [Serra et al., 2018] learns a hard attention mask for each task to preserve the knowledge of previous tasks by freezing a portion of the weights
  - PNN [Rusu et al., 2016] instantiates new networks incrementally for each new task and adds lateral connections to previous knowledge
  - Increase in network complexity and growth in memory requirement

Progressive Neural Network with 3 tasks



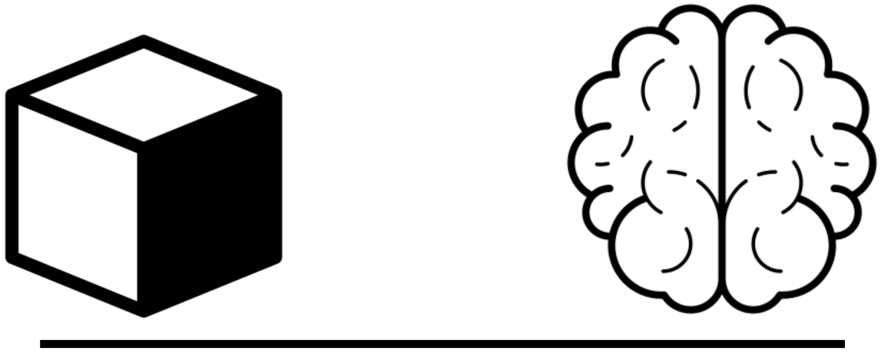
[Rusu et al., 2016]

Stability-Plasticity Dilemma in CL

Catastrophic forgetting: significant performance degradation on old tasks when new tasks are learnt

- Updates override knowledge learnt from previous tasks
- Overridden knowledge cannot be recovered without available data from previous tasks

Maintain old knowledge   Learn new knowledge



Stability   Plasticity

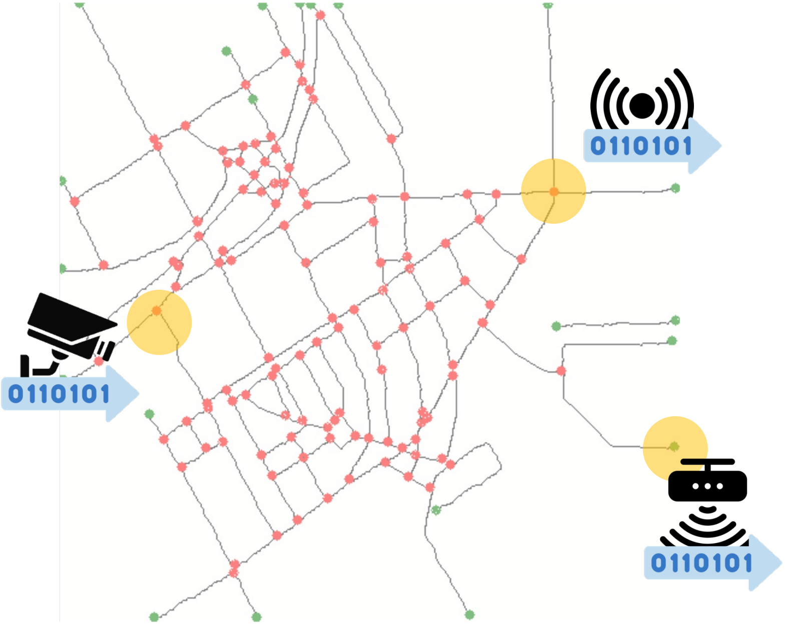
Bridging PFL + CL

- Data stationarity is a common assumption in PFL
- However, changes in the underlying data distributions over time are expected in dynamic real-world systems

Goal of PCFL: train PFL models on changing data distributions over time

Alibaba City Brain: Traffic forecasting & urban planning

[Alibaba DAMO, 2022]



FL: privacy-preserving collaborative learning
PFL: personalized model for local adaptation
CL: learning without forgetting on big data streams

NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Personalized Continual Federated Learning (PCFL)

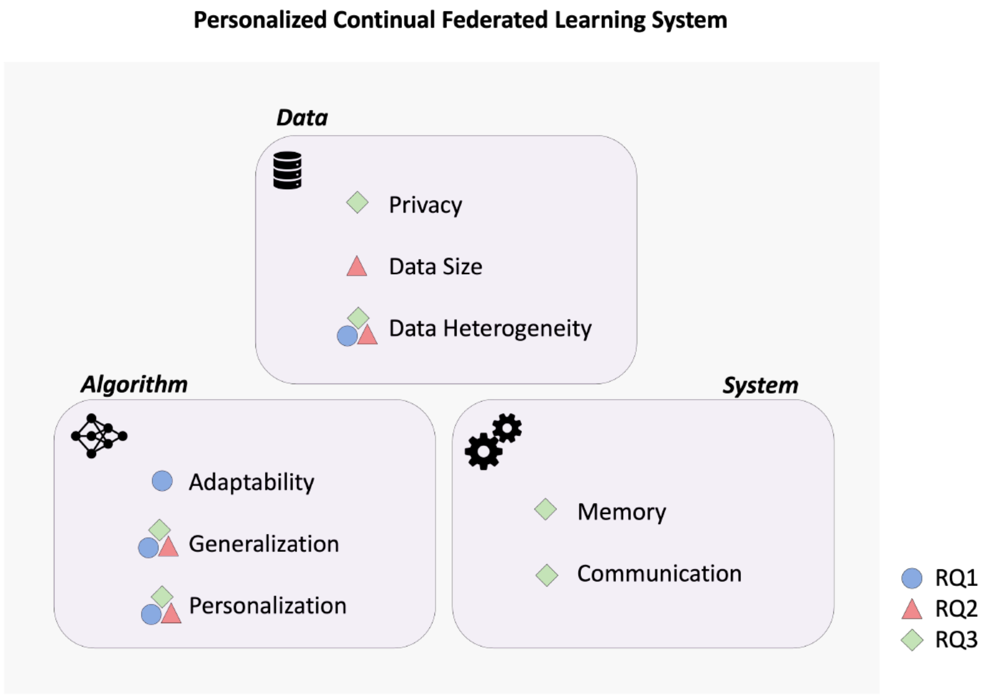


Figure 1.1: Challenges addressed by each open research question in Personalized Continual Federated Learning systems.

RQ1 : How to incrementally adapt an existing trained PCFL model to newly collected local data?

RQ2 : How to train PCFL models in few-shot settings?

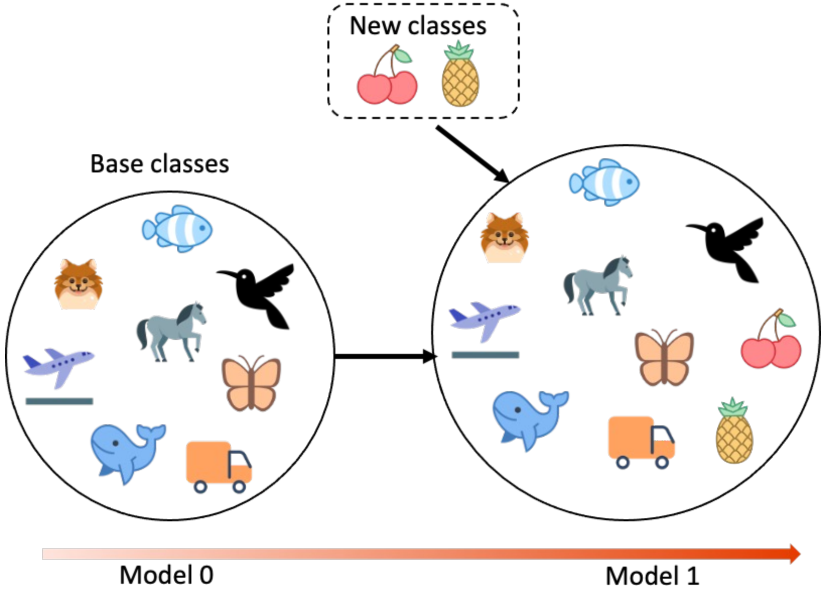
RQ3 : How to achieve memory and communication efficiency in PCFL?

Research Directions

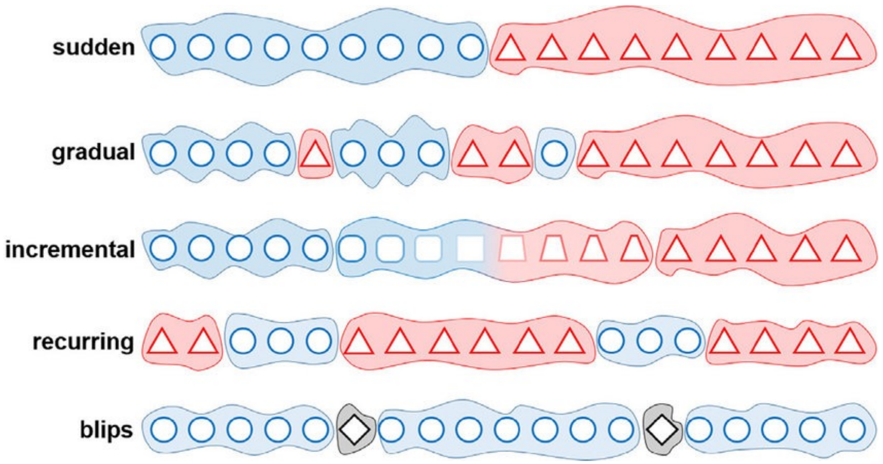
RQ1: How to incrementally adapt an existing trained PCFL model to newly collected local data?

- In deployed FL systems, there are often changes in the underlying data distributions
- Example: adapting the FL model to a new target market
    - New target classes, different data distributions

Expanding target classes



Distribution shifts



Research Directions

RQ2: How to train PCFL models in few-shot settings?

- Data scarcity (lack of quality training data) is the key motivation for clients who join FL
- Challenges
  - Avoid forgetting on old classes
  - Prevent overfitting to few-shot data of new classes

Research Directions

RQ3: How to achieve memory and communication efficiency in PCFL?

- FL client devices have significant variability in hardware capabilities in terms of memory, power, network connectivity
- A memory budget is required in many CL approaches, which is not applicable to memory constrained client devices
- Potential privacy risks from long-term data storage
- Need for communication-efficient mechanisms to address bandwidth challenges

Thank you!