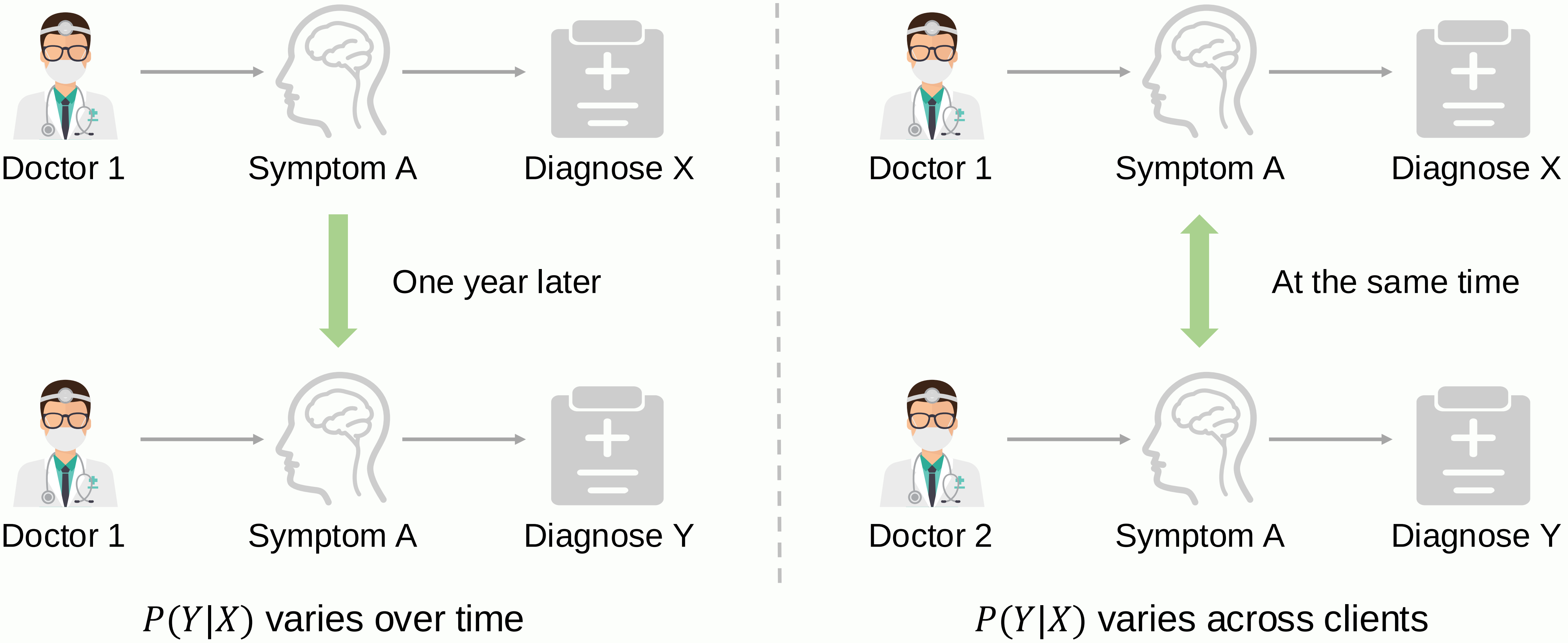
Classifier Clustering and Feature Alignment for Federated Learning under Distributed Concept Drift

Junbao Chen, Jingfeng Xue, Yong Wang, Zhenyan Liu*, Lu Huang

School of Computer Science and Technology, Beijing Institute of Technology, Beijing, China

Code: https://github.com/Chen-Junbao/FedCCFA

Distributed Concept Drift



These two types of drift may **occur simultaneously** in federated learning

Motivation

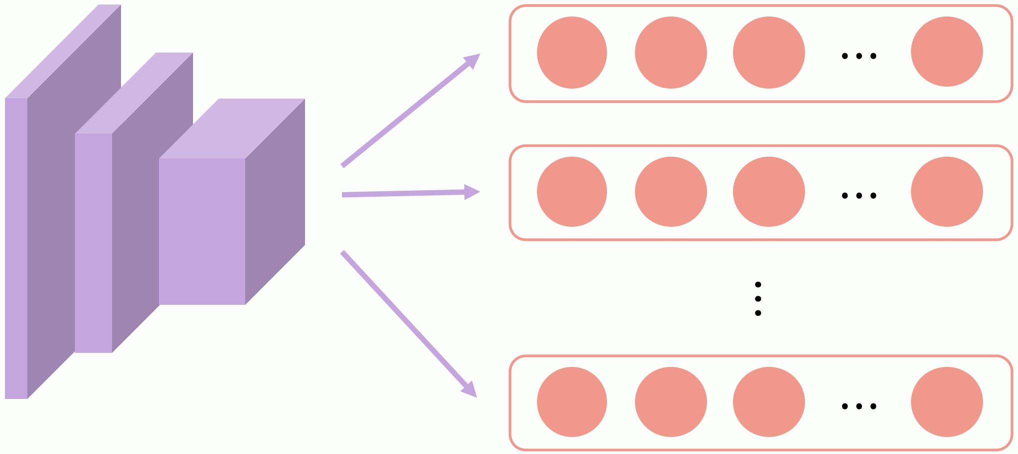
- A single model fails to adapt to distributed concept drift:

Client 1: $f(x_1;\theta) = y_1$

Client 2: $f(x_1;\theta) = y_2$

Impossible

- Marginal distribution $P(X)$ is invariant when drift occurs



Personal $P_1(Y|X)$

Personal $P_2(Y|X)$

$\vdots$

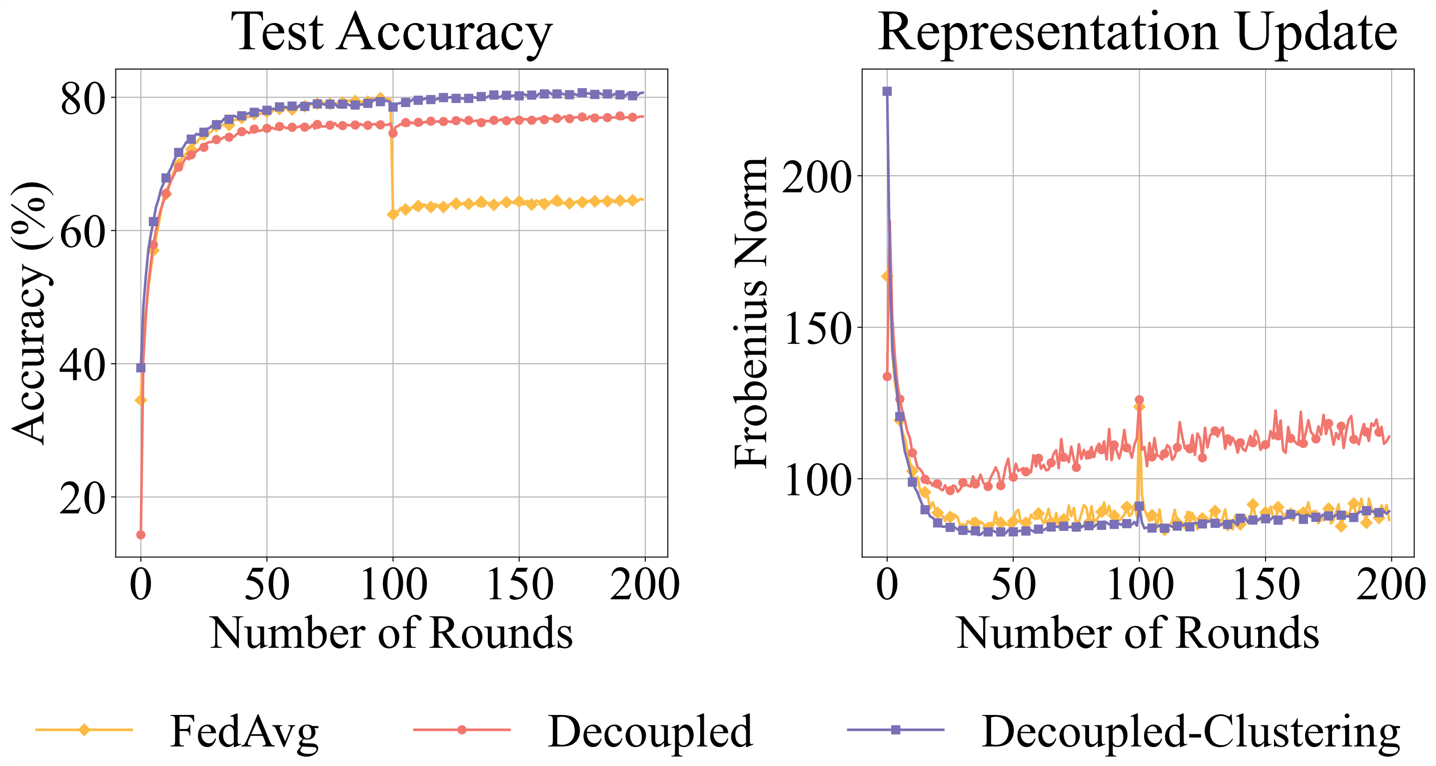
Personal $P_k(Y|X)$

Invariant $P(X)$

Feature extractor can be shared across clients, and each client owns a local classifier

Classifier Clustering

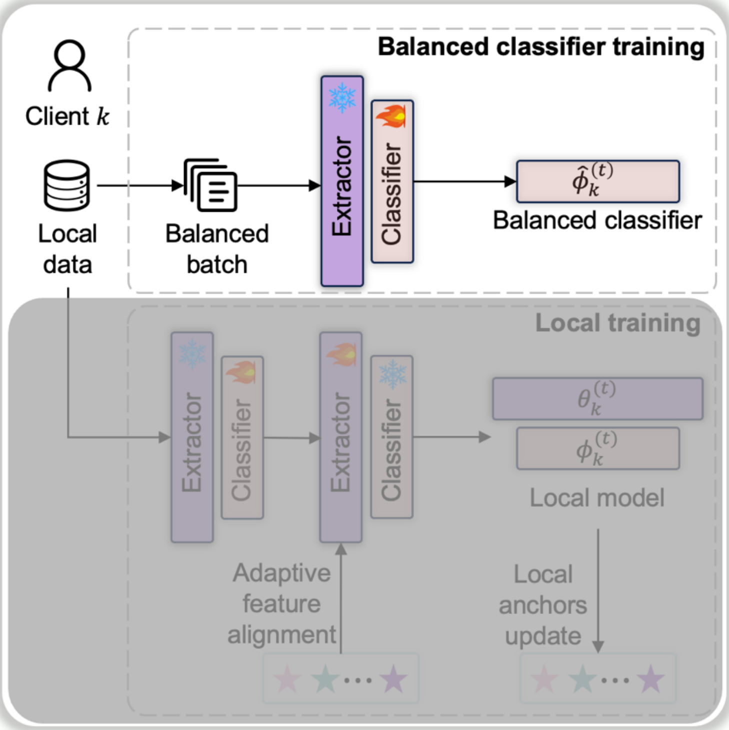
- Some clients may share the same conditional distribution $P(Y|X)$
- Merging the classifiers under the same $P(Y|X)$ can improve the generalization performance



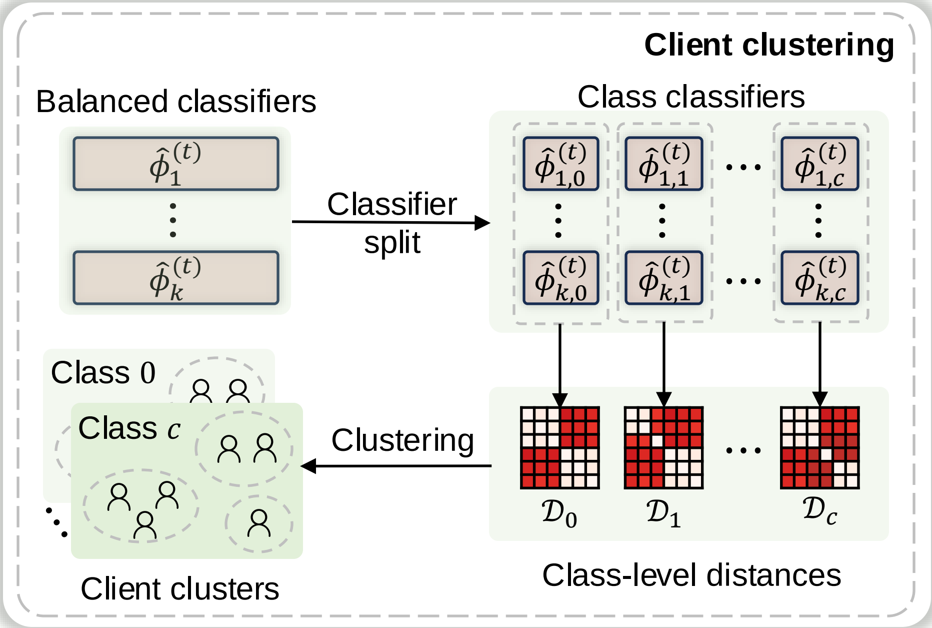
Decoupled-Clustering adapts to distributed concept drift and achieves **better performance**

Classifier Clustering

- Train classifier using a balanced batch



- Separate the classifier for each class and merge the classifiers under the same $P(Y|X)$



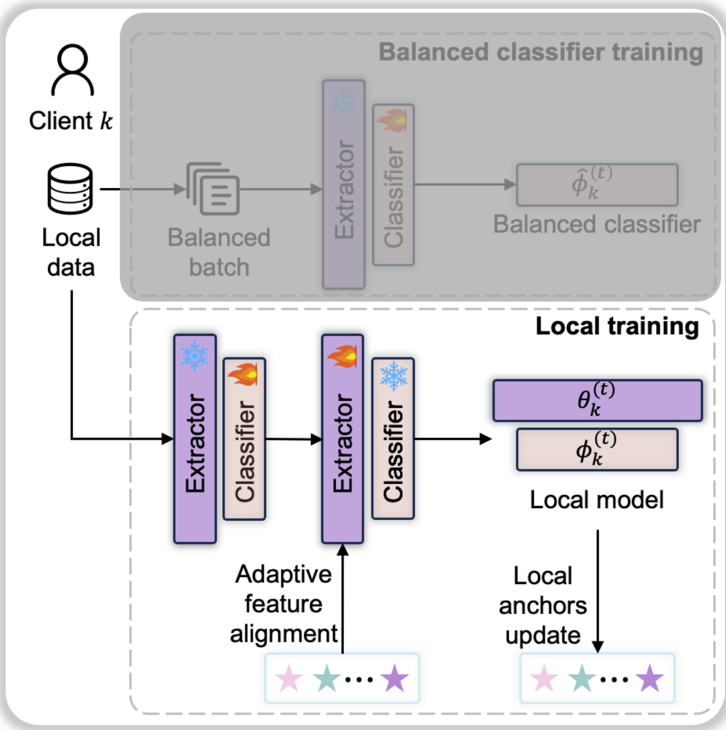
- **Why balanced batch?**
  The classifiers can be easily biased towards head classes. Train the local classifier using a balanced batch to solve this disturbance by class imbalance.

$$D_c(i,j) = \frac{1}{\left|I^{(t)}\right| - 2} \sum_{q \in I^{(t)}\{i,j\}} \left| Cos\left(\hat{\phi}_{i,c}^{(t)}, \hat{\phi}_{q,c}^{(t)}\right) - Cos\left(\hat{\phi}_{j,c}^{(t)}, \hat{\phi}_{q,c}^{(t)}\right) \right|, \forall i,j \in I^{(t)}$$

MADD: [1] Soham Sarkar and Anil K Ghosh. On perfect clustering of high dimension, low sample size data. IEEE transactions on pattern analysis and machine intelligence, 42(9):2257–2272, 2019.

Adaptive Feature Alignment

- Contrastive-guiding feature alignment

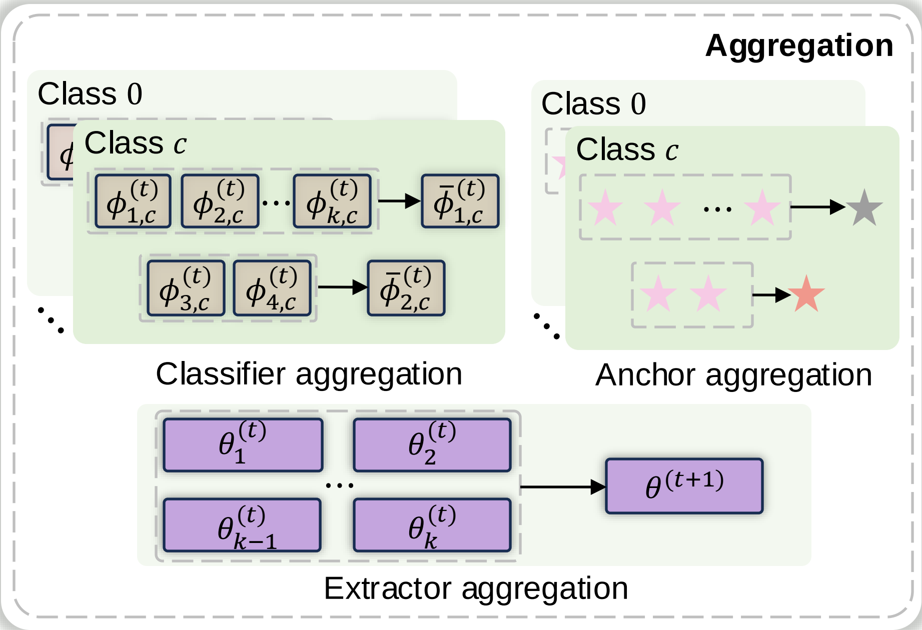


Adjust alignment weight with the **entropy of $P(Y)$**

$$G_k\left(\theta_k^{(t)}; A_k^{(t)}\right) = -\log \frac{\exp\left(sim\left(f_{\theta_k^{(t)}}(x), A_{k,c}^{(t)}/\tau\right)\right)}{\sum_{i=1}^{C} \exp\left(sim\left(f_{\theta_k^{(t)}}(x), A_{k,i}^{(t)}/\tau\right)\right)}$$

$$L_k(\theta, \phi_k) = F_k(\theta, \phi_k) + \frac{H\left(P_k^{(t)}(Y)\right)}{\gamma}$$

- Aggregation of classifiers, extractors and anchors



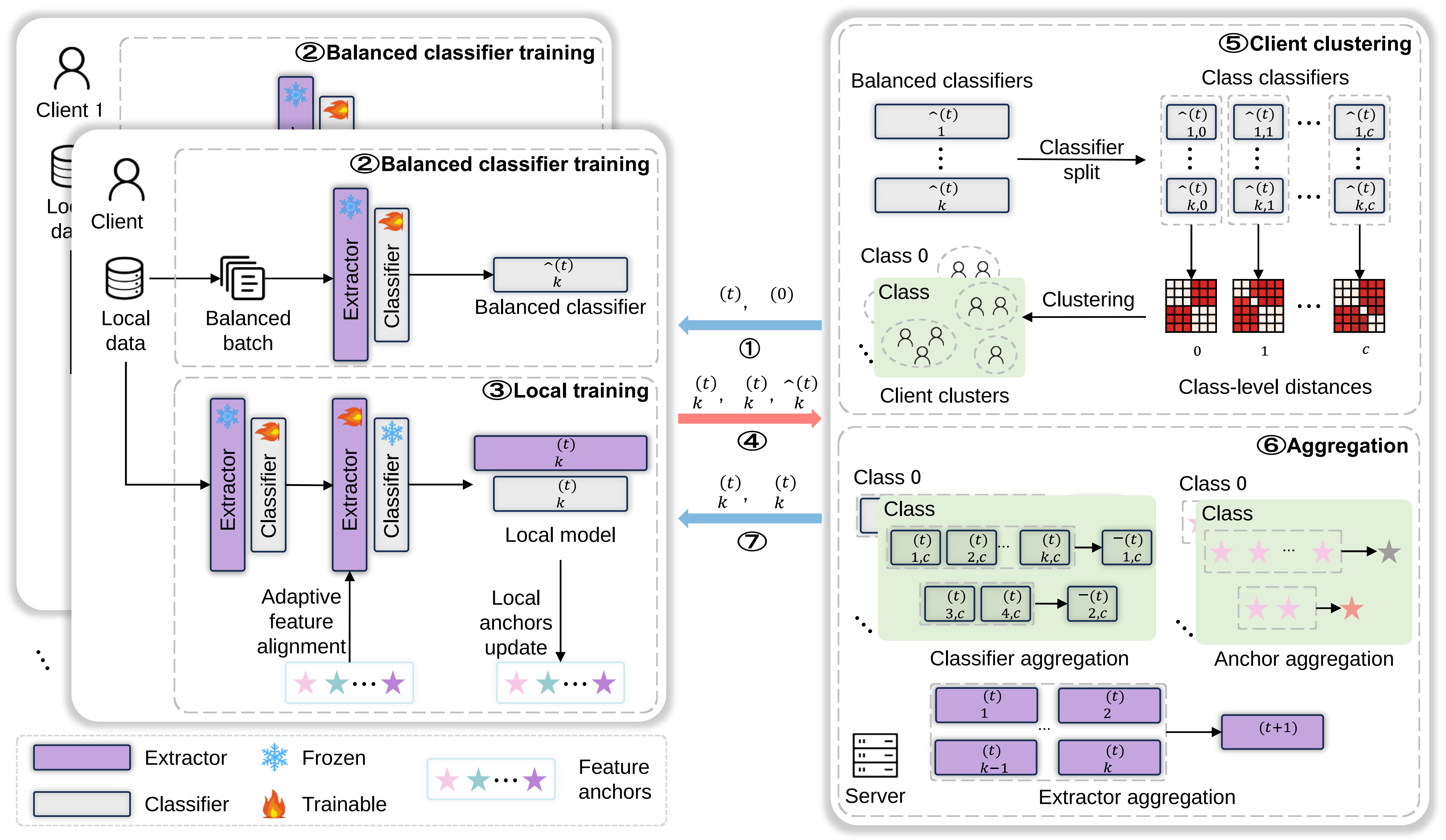
$$\phi_{m,c} = \frac{1}{\left|S_{m,c}^{(t)}\right|} \sum_{k \in S_{m,c}^{(t)}} \phi_{k,c}^{(t)}, \forall S_{m,c}^{(t)} \in S_c^{(t)}$$

$$A_{m,c}^{(t)} = \frac{1}{\left|S_{m,c}^{(t)}\right|} \sum_{k \in S_{m,c}^{(t)}} a_{k,c}^{(t)}, \forall S_{m,c}^{(t)} \in S_c^{(t)}$$

$$\theta^{(t+1)} = \frac{1}{\sum_{k \in I^{(t)}} \left|D_k^{(t)}\right|} \sum_{k \in I^{(t)}} \left|D_k^{(t)}\right| \theta_k^{(t)}$$

Clients in the same cluster $k \in S_{m,c}^{(t)}$ share
the clustered classifiers and anchors

FedCCFA

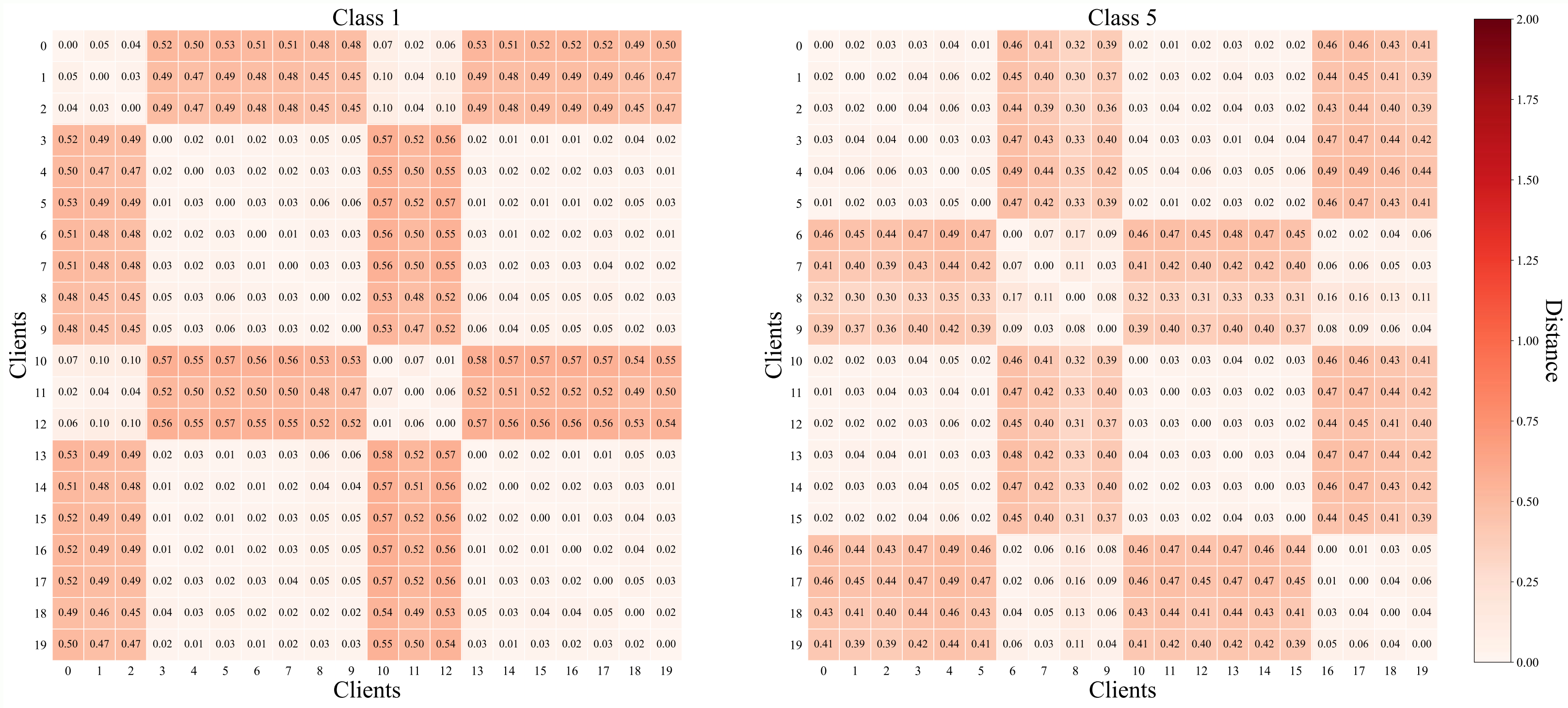

Client 1
②Balanced classifier training
Client
Local data
Balanced batch
Extractor
Classifier
Balanced classifier
③Local training
Local model
Adaptive feature alignment
Local anchors update
Extractor
Frozen
Feature anchors
Classifier
Trainable
①
④
⑦
⑤Client clustering
Balanced classifiers
Class classifiers
Classifier split
Class 0
Class
Clustering
Client clusters
Class-level distances
⑥Aggregation
Class 0
Class
Classifier aggregation
Anchor aggregation
Server
Extractor aggregation

Generalized Accuracy under Sudden Drift

Method	Fashion-MNIST		CIFAR-10		CINIC-10	
	20 clients	100 clients	20 clients	100 clients	20 clients	100 clients
FedAvg	67.81±1.12	68.15±1.98	60.96±0.37	58.15±0.23	49.69±0.16	46.18±0.39
FedProx	69.25±0.60	68.47±0.19	61.25±0.20	57.68±0.11	49.45±0.40	46.11±0.59
SCAFFOLD	69.92±1.24	69.66±0.24	58.67±0.39	45.87±1.29	46.34±0.70	37.94±1.43
FedFM	68.48±0.79	69.12±0.34	60.61±0.27	57.51±0.35	50.25±0.11	46.23±0.81
AdapFedAvg	69.30±1.06	68.86±0.53	60.92±0.20	58.23±0.30	49.86±0.36	46.34±0.19
Flash	10.00±0.00	71.16±0.28	59.84±0.75	60.49±0.27	49.44±0.41	49.28±0.40
pFedMe	82.37±0.09	77.66±0.17	58.92±1.64	44.15±1.08	41.34±0.22	37.76±0.78
Ditto	78.51±0.82	79.62±0.31	67.45±0.01	63.35±0.67	51.06±0.26	48.40±0.34
FedRep	81.99±0.37	81.07±0.29	64.51±0.86	53.30±1.51	44.00±0.26	38.15±0.14
FedBABU	79.83±0.18	81.68±0.10	58.87±0.50	55.29±0.61	41.24±0.90	40.49±0.30
FedPAC	84.00±0.47	87.24±0.16	66.47±0.30	64.73±1.22	44.38±0.18	46.22±0.47
IFCA	87.02±0.06	88.25±0.21	61.24±0.32	57.31±0.44	44.04±2.00	45.43±0.22
FedDrift	78.95±1.30	87.75±0.04	48.17±2.43	57.26±0.36	34.23±0.04	45.60±1.07
FedCCFA	**89.50±0.10**	**89.39±0.14**	**76.95±0.63**	**73.21±0.82**	**51.12±1.70**	**52.62±0.51**

FedCCFA achieves **SOTA performance**

Classifier Clustering Results



Clustering input	CIFAR-10
local classifiers	68.39±0.71
balanced classifiers	73.21±0.82
oracle	73.55±0.27

- Our clustering method **effectively measures the distance** between each client
- Classifier clustering significantly **improves generalization performance**

Adaptive Feature Alignment Results

Alignment methods	CIFAR-10
w/o alignment	71.40±0.43
w/o clustered	72.77±0.71
w/ clustered	73.21±0.82

Weight	$Dir(0.01)$	$Dir(0.1)$	$Dir(0.5)$
fixed (0)	62.46±0.92	69.33±1.47	72.78±0.53
fixed (0.1)	57.86±1.91	70.72±0.83	73.83±0.53
fixed (1.0)	40.89±3.97	67.43±1.08	74.80±0.41
$\gamma = 10$	60.30±0.79	69.75±0.19	74.97±0.48
$\gamma = 20$	61.29±0.59	70.21±0.48	74.44±0.26
$\gamma = 50$	62.15±0.49	70.11±1.05	73.32±0.06

Clustered feature anchors facilitate **more precise feature alignment** under drift
Adaptive alignment weight is **robust to the degree of heterogeneity**

北京理工大學
BEIJING INSTITUTE OF TECHNOLOGY

NEURAL INFORMATION PROCESSING SYSTEMS

Thank You!

Classifier Clustering and Feature Alignment for Federated Learning under Distributed Concept Drift

Junbao Chen, Jingfeng Xue, Yong Wang, Zhenyan Liu*, Lu Huang

https://arxiv.org/abs/2410.18478

https://github.com/Chen-Junbao/FedCCFA