Front. Comput. Sci., 2025, 19(6): 196339
https://doi.org/10.1007/s11704-024-40557-w

LETTER

A federated anti-forgetting representation method based on hybrid model architecture and gradient truncation

Hui WANG (✉)[1], Jie SUN[2], Tianyu WO (✉)[2,3], Xudong LIU[1,2], Suzhen PEI[4]

1 School of Computer Science and Engineering, Beihang University, Beijing 100191, China
2 Zhongguancun Laboratory, Beijing 100194, China
3 School of Software, Beihang University, Beijing 100191, China
4 School of Mathematics and Statistics, Southwest University, Chongqing 400715, China

1 Introduction

Unsupervised Federated Continual Learning (UFCL) is a new learning paradigm that embeds unsupervised representation techniques into the Federated Learning (FL) framework, which enables continuous training of a shared representation model without compromising individual participants' data privacy [1,2]. However, the continuous learning process may cause catastrophic forgetting in the model, reducing generated representations' performance.

Our research findings suggest that limited model capacity and undifferentiated weight aggregation in UFCL are mainly responsible for decreased model performance. Therefore, this paper proposes an anti-forgetting representation learning method based on a new hybrid model architecture and gradient truncation technique, namely FedAFR. The contributions can be summarized as follows. (1) We propose a model architecture based on *Kolmogorov-Arnold* [3] and pluggable structures, which can effectively improve the model's memory capacity and anti-forgetting ability. (2) We design a gradient truncation technique to reduce the interference of weight aggregation on model memory and use an ordinary differential equation (ODE) sampler [4] to augment the representation performance. (3) We carry out experiments to compare FedAFR against the state-of-the-art representation methods in FL.

2 Methodology

A UFCL scenario typically contains a global coordinator and P participants and uses contrastive techniques [5] to extract the representation of the input sample. Let $M(\cdot;\theta_p^e)$ denote the local model of participant p in the eth $\in[E]$ optimization round. We denote by L_p^c the contrastive loss of $M(\cdot;\theta_p^e)$. The objective of FedAFR is formally defined as:

$$\arg\min_{\theta} \frac{1}{EP}\sum_{e=1}^{E}\sum_{p=1}^{P}\Big[L_p^c\Big(\mathcal{M}(x_p^e;\theta_p^e)\Big)+ \alpha\sum_{j<e}KL\Big(\mathcal{M}(x_p^e;\theta_p^e),\mathcal{M}(x_p^j;\theta_p^j)\Big)\Big], \quad (1)$$

where $KL(\cdot)$ denotes the *Kullback-Leibler* divergence [6], it encourages $M(\cdot;\theta_p^e)$ to have a similar performance to $M(\cdot;\theta_p^j)$ in any historical round j, i.e., maintaining memories in any historical rounds. α is the forgetting penalty coefficient.

2.1 Hybrid federated model architecture

Our model comprises an input layer, an output layer, multiple blocks $\{\mathcal{B}_b\}_{b=1}^B$, and a diffusion module $\mathcal{D}$ (Fig. 1). All the blocks and diffusion module share the input, and the output is a weighted mixture of all blocks' output, i.e., $\sum_{b=1}^B \omega_b\mathcal{B}_b(x)$, where B represents the block number. Each block is composed of a KAU and a pluggable unit (PU) such as CNN or ResNet18 connected in series. KAU has learnable activation functions at the edges, and weight is replaced by a univariate function parameterized as a spline function. We denote by K_b^L with the shape $[n_0,n_1,\ldots,n_L]$ the L-layers KAU in the block $\mathcal{B}_b$, where n_l is the number of nodes in the lth layer. We denote by a matrix of 1D functions $\Phi=\{\phi_{i,j},\ i\in[I],\ j\in[O]\}$ the layer with I dimensional input and O dimensional output, where the functions $\phi_{i,j}$ have trainable parameters. In the lth layer of K_b^L, we denote the ith node and its corresponding activation value by (l,i) and $x_{l,i}$. There are $n_l\times n_{l+1}$ activation functions between layer l and layer $l+1$, which denoted by $\{\phi_{l,j,i},\ i\in[n_l]\ ,j\in[n_{l+1}]\}$. We denote by $x_{l,i}$ and $\phi_{l,j,i}(x_{l,i})$ the pre-activation and post-activation of $\phi_{l,j,i}$, so the activation value of the $(l+1,j)$ node is $x_{l+1,j}=\sum_{i=1}^{n_l}\phi_{l,j,i}(x_{l,i}),\ j\in[n_{l+1}]$. In summary, the output of K_b^L is $(\Phi_{L-1}\circ\Phi_{L-2}\circ\cdots\circ\Phi_0)\cdot\mathbf{x}$, where Φ_l is the function matrix corresponding to the lth layer of K_b^L, that is

Received June 3, 2024; accepted September 9, 2024

E-mail: whui@buaa.edu.cn; woty@buaa.edu.cn

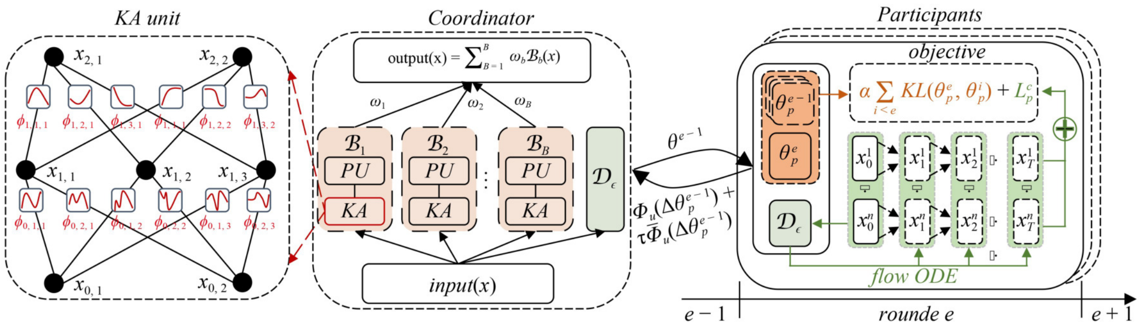

Fig. 1 The overview of our proposed FedAFR

$$x_{l+1} = \underbrace{\begin{pmatrix} \phi_{l,1,1}(\cdot) & \phi_{l,1,2}(\cdot) & \cdots & \phi_{l,1,n_l}(\cdot) \\ \phi_{l,2,1}(\cdot) & \phi_{l,2,2}(\cdot) & \cdots & \phi_{l,2,n_l}(\cdot) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_{l,n_{l+1},1}(\cdot) & \phi_{l,n_{l+1},2}(\cdot) & \cdots & \phi_{l,n_{l+1},n_l}(\cdot) \end{pmatrix}}_{\mathbf{\Phi}_l} x_l .$$

2.2 High-order ODE sampler for representation

We use the diffusion module $\mathcal{D}$ to augment the participants' samples. Sampling from $\mathcal{D}$ can be viewed as solving a diffusion ODE [4]. Specifically, the forward *Markov* process in $\mathcal{D}$ continues to inject *Gaussian* noise into sample $x_0 \sim q(x_0)$ in T steps and generates a sequence of variables $\{x_1, ..., x_T\}$, forming a mapping from the $q(x_0)$ to the normal distribution $\mathcal{N}(\mathbf{0}, \mathbf{I})$. The distribution of $x_t, t \in [T]$ satisfies $q_{0-T}(x_t|x_0) = \mathcal{N}(x_t|\alpha_t x_0,\ \delta_t^2 \mathbf{I})$, where α_t and δ_t represent the noise sequence. This transition can be equivalently expressed using stochastic differential equation (SDE) $dx_t = f(t)x_t dt + g(t)dw_t, f(t) = \frac{d\log\alpha_t}{dt}, g^2(t) = \frac{d\delta_t^2}{dt} - 2\frac{d\log\alpha_t}{dt}\alpha_t^2$, where w_t is the *Wiener* process [4]. The reverse *Markov* process (RMP) in $\mathcal{D}$ can be formalized as $dx_t = \left[f(t)x_t - g^2(t)\nabla_x \log q_t(x_t)\right]dt + g(t)d\hat{w}_t$, $x_T \sim q_T(x_T)$ and $\hat{w}_t$ is a reverse *Wiener* process. The above RMP has an equivalent flow ODE $\frac{dx_t}{dt} = f(t)x_t - \frac{1}{2}g^2(t)\nabla_x \log q_t(x_t)$ [4]. We denote the $\mathcal{D}$ by $\mathcal{D}_\epsilon(x_t, t)$ and convert the above flow ODE into $\frac{dx_t}{dt} = f(t)x_t + \frac{g^2(t)}{2\delta_t}\mathcal{D}_\epsilon(x_t, t)$, $x_T \sim \mathcal{N}(\mathbf{0}, \hat{\delta}^2\mathbf{I})$ [7], with the solution $x_t = \frac{\alpha_t}{\alpha_s}x_s - \alpha_t \int_{\lambda_s}^{\lambda_t} e^{-\lambda}\mathcal{D}_\epsilon(x_\lambda, \lambda)d\lambda$, where $0 \leqslant t \leqslant s$ [7]. From *Taylor* equation, we have $\mathcal{D}_\epsilon(x_\lambda, \lambda) = \sum_{n=0}^{k-1} \frac{(\lambda - \lambda_{t_i - 1})^n}{n!}\mathcal{D}_\epsilon^{(n)}(x_{\lambda_{t_i-1}}, \lambda_{t_i-1}) + \mathcal{O}((\lambda - \lambda_{t_i-1})^k)$. Substituting the expansion into solution yields $(k = 2)$:

$$x_{t_i} = \frac{\alpha_{t_i}}{\alpha_{t_{i-1}}}x_{t_{i-1}} - \delta_{t_i}(e^{\lambda_{t_i} - \lambda_{t_{i-1}}} - 1)\mathcal{D}_\epsilon\left(v_i,\ u_i\right), \qquad (2)$$

where $u_i = t_\lambda\left(\frac{\lambda_{t_{i-1}} + \lambda_{t_i}}{2}\right)$ and $v_i = \frac{\alpha_{u_i}}{\alpha_{t_{i-1}}}x_{t_{i-1}} - \delta_{u_i}\left(e^{\frac{\lambda_{t_i} - \lambda_{t_{i-1}}}{2}} - 1\right)\mathcal{D}_\epsilon(x_{t_{i-1}},\ t_{i-1})$.

2.3 Gradient truncation in weight aggregation

The coordinator aggregates the gradients of all participants using the following gradient truncation:

$$\theta^e \leftarrow \theta^{e-1} + \sum_{p=1}^{P}\left(\Psi_u(\Delta\theta_p^{e-1}) + \tau\,\overline{\Psi}_u(\Delta\theta_p^{e-1})\right), \qquad (3)$$

where $\Delta\theta_p^{e-1}$ denotes the gradients of participant p in optimization round e. $\Psi_u(\cdot)$ denotes taking the top $u \in (0,1)$ gradients with the most remarkable change, $\overline{\Psi}_u(\cdot)$ represents the remaining gradients, and $\tau \in [0,1)$ denotes the truncation coefficient. Equation (3) reduces interference on model memory by weakening tail gradients in weight aggregation.

3 Experiment and analysis

We compare FedAFR with SimSiam [5], RELIC, FedCLR [8], FedCA [9], and FedWeIT [10] over Accuracy and Forgetting metrics. Table 1 shows FedAFR outperforms baselines by 7.8% on average accuracy while forgetting has an average decrease of 16.1%. The comparison indicates that the hybrid architecture and gradient truncation effectively improve model representation and anti-forgetting performance. Table 2 describes the impact of u on forgetting at $\tau = 0.5$. With the decrease of u, forgetting shows a decreasing trend. Reducing u means the weight scope involved in aggregation decreases, diminishing weight interference. However, a further decrease in u will result in the model losing more valuable weight information, leading to growth in the optimization round.

4 Conclusion

This paper formulates the anti-forgetting representation problem under UFCL and proposes FedAFR. The experiments show that FedAFR effectively improves the model's anti-forgetting performance.

Acknowledgements This work was supported by the National Science and Technology Major Project (2022ZD0120203).

Table 1 Comparison of the model accuracy and forgetting ($u = 0.5, \tau = 0.5$)

Method	CMNIST		CCIFAR10		FFHQ		MiniImageNet	
	Acc ↑	Forgetting ↓	Acc ↑	Forgetting ↓	Acc ↑	Forgetting ↓	Acc ↑	Forgetting ↓
SimSiam	87.73 ± 0.13	13.39 ± 1.23	49.70 ± 1.24	10.09 ± 1.02	59.17 ± 0.36	09.42 ± 0.87	82.57 ± 0.83	07.12 ± 1.62
RELIC	86.36 ± 0.62	12.01 ± 0.43	48.11 ± 0.78	08.11 ± 0.33	60.61 ± 0.12	08.11 ± 0.53	81.63 ± 0.62	08.01 ± 1.22
FedCLR	88.46 ± 0.41	10.11 ± 1.85	51.06 ± 0.78	07.51 ± 0.42	60.09 ± 0.78	10.22 ± 0.85	83.17 ± 0.49	09.11 ± 1.07
FedCA	87.63 ± 0.89	10.84 ± 0.87	50.61 ± 0.73	06.21 ± 0.53	59.37 ± 0.79	09.81 ± 0.67	84.17 ± 0.86	08.07 ± 1.71
FedWeIT	89.16 ± 0.76	06.09 ± 0.31	52.12 ± 0.79	03.71 ± 0.43	63.77 ± 0.82	04.62 ± 0.61	85.71 ± 0.62	04.11 ± 0.12
FedAFR	**90.52** ± 0.41	**03.11** ± 0.08	**53.67** ± 0.56	**03.11** ± 0.13	**65.05** ± 0.43	**04.32** ± 0.31	**88.41** ± 0.26	**02.21** ± 0.13

Table 2 The impact of u on forgetting

u	FFHQ			MiniImageNet		
	Round	Acc	Forgetting	Round	Acc	Forgetting
0.9	393	64.93 ± 1.1	09.74 ± 0.8	452	87.25 ± 1.2	09.21 ± 0.5
0.7	481	65.07 ± 1.3	07.98 ± 0.7	582	87.31 ± 1.1	06.17 ± 0.2
0.5	559	65.16 ± 1.8	04.27 ± 0.3	691	88.31 ± 0.7	02.27 ± 0.6
0.3	701	66.21 ± 1.1	03.13 ± 0.2	759	88.66 ± 1.3	02.23 ± 0.7
0.1	952	67.85 ± 1.3	02.23 ± 0.1	884	89.01 ± 0.8	01.76 ± 0.9

Competing interests The authors declare that they have no competing interests or financial conflicts to disclose.

References

1. Liu F, Zheng Z, Shi Y, Tong Y, Zhang Y. A survey on federated learning: a perspective from multi-party computation. Frontiers of Computer Science, 2024, 18(1): 181336
2. Wang H, Sun J, Wo T, Liu X. FED-3DA: a dynamic and personalized federated learning framework. In: Proceedings of 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). 2023, 1–5
3. Liu Z, Wang Y, Vaidya S, Ruehle F, Halverson J, Soljačić M, Hou T Y, Tegmark M. KAN: Kolmogorov-Arnold networks. 2024, arXiv preprint arXiv: 2404.19756
4. Lu C, Zhou Y, Bao F, Chen J, Li C, Zhu J. DPM-solver: a fast ODE solver for diffusion probabilistic model sampling in around 10 steps. In: Proceedings of the 36th International Conference on Neural Information Processing Systems. 2022, 418
5. Chen X, He K. Exploring simple Siamese representation learning. In: Proceedings of 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2021, 15745–15753
6. Ren L, Jiang L, Zhang W, Li C. Label distribution similarity-based noise correction for crowdsourcing. Frontiers of Computer Science, 2024, 18(5): 185323
7. Li Q, Li G, Niu W, Cao Y, Chang L, Tan J, Guo L. Boosting imbalanced data learning with wiener process oversampling. Frontiers of Computer Science, 2017, 11(5): 836-851
8. Chen T, Kornblith S, Norouzi M, Hinton G. A simple framework for contrastive learning of visual representations. In: Proceedings of the 37th International Conference on Machine Learning. 2020, 1597–1607
9. Zhang F, Kuang K, Chen L, You Z, Shen T, Xiao J, Zhang Y, Wu C, Wu F, Zhuang Y, Li X. Federated unsupervised representation learning. Frontiers of Information Technology & Electronic Engineering, 2023, 24(8): 1181-1193
10. Yoon J, Jeong W, Lee G, Yang E, Hwang S J. Federated continual learning with weighted inter-client transfer. In: Proceedings of the 38th International Conference on Machine Learning. 2021, 12073–12086