Toward Enhancing Representation Learning in Federated Multi-Task Settings

Mehdi Setayesh, Mahdi Beitollahi, Yasser H. Khalil, and Hongliang Li

Huawei Noah's Ark Lab, Montreal, Canada

April 2026

Introduction

- In practical federated multi-task learning (FMTL) settings, users may have
  - Heterogeneous tasks;
  - Task-specific local datasets;
  - Heterogeneous model architectures.

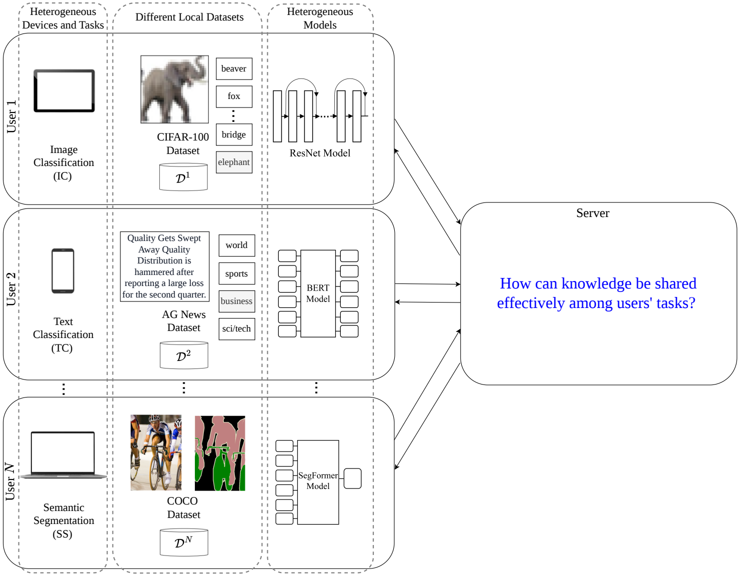
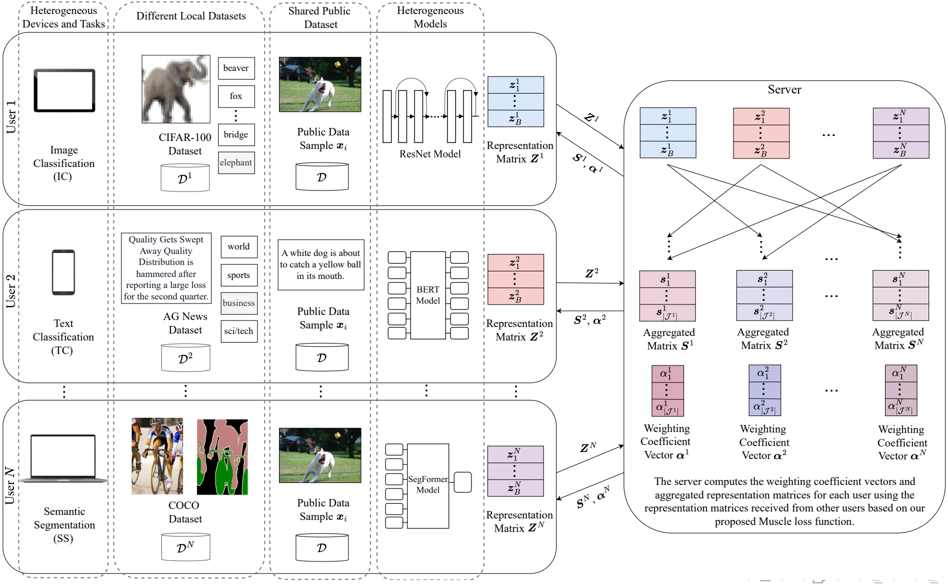


Figure: Illustration of a practical FMTL setting.

CONTRIBUTIONS

- We reframe the FMTL objective to focus on learning a shared representation space across tasks instead of shared model parameters.
  - This addresses the model congruity assumption inherent in most existing FMTL approaches.
- We propose a novel contrastive learning (CL) loss named *Muscle* (**Mu**lti-task/modal **S**ystematic **C**ontrastive **Le**arning).
  - Compared to pairwise alignment, Muscle loss effectively captures dependencies among the representations of all models in multi-model settings.
- We design a novel FMTL algorithm named *FedMuscle* by leveraging the proposed Muscle loss.
  - In FedMuscle, users transmit their local knowledge derived from a shared public dataset to the server, rather than sharing local data or model parameters.

The FedMuscle Framework



Proposed Contrastive Loss: Muscle

- Taking $\{\boldsymbol{z}_i^m\}_{m=1,\, m\neq n}^N$ as positives for the anchor $\boldsymbol{z}_i^n$, the Muscle loss is defined as follows:

$$\mathcal{L}^n_{\text{Muscle}}(\boldsymbol{z}_i^n) = -\log \frac{f(\boldsymbol{z}_i^n,\, \{\boldsymbol{z}_i^m\}_{m=1,\, m\neq n}^N)}{\sum_{\boldsymbol{j}\in\mathcal{J}^n} f(\boldsymbol{z}_i^n,\, \{\boldsymbol{z}_{j_m}^m\}_{m=1,\, m\neq n}^N)},$$

where $\mathcal{J}^n = \{\boldsymbol{j} = (j_1, \ldots, j_N) \,|\, j_m \in [B],\, m \in [N] \setminus \{n\}\}$.

- The Muscle loss is obtained as follows:

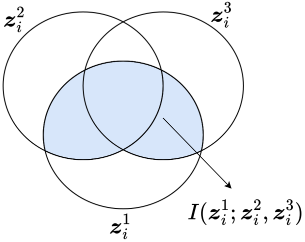
$$\boxed{\mathcal{L}^n_{\text{Muscle}}(\boldsymbol{z}_i^n) = -\log \frac{\alpha_{(i,\ldots,i)} \exp(\boldsymbol{z}_i^n \cdot \sum_{m\in[N]\setminus\{n\}} \boldsymbol{z}_i^m / \tau_{n,m}^{(N)})}{\sum_{\boldsymbol{j}\in\mathcal{J}^n} \alpha_{\boldsymbol{j}} \exp(\boldsymbol{z}_i^n \cdot \sum_{m\in[N]\setminus\{n\}} \boldsymbol{z}_{j_m}^m / \tau_{n,m}^{(N)})}},$$

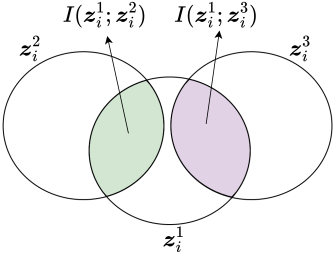
where $\tau_{n,m}^{(N)}$ is a temperature parameter, $\alpha_{\boldsymbol{j}} = \exp(-\frac{1}{2} \sum_{m\in[N]\setminus\{n\}} \sum_{m'\in[N]\setminus\{n,m\}} \gamma_{m,m'}^{(N)} \boldsymbol{z}_{j_m}^m \cdot \boldsymbol{z}_{j_{m'}}^{m'})$, and $\gamma_{m,m'}^{(N)} = 1/\tau_{m,m'}^{(N-1)} - 1/\tau_{m,m'}^{(N)}$.

Muscle Loss: A Mutual Information Perspective

- Given $\mathcal{L}^n_{\text{Muscle}}(\boldsymbol{z}_i^n)$, the mutual information $I(\boldsymbol{z}_i^n; \{\boldsymbol{z}_i^m\}_{m=1,\, m\neq n}^N)$ is lower-bounded as follows:

$$I(\boldsymbol{z}_i^n; \{\boldsymbol{z}_i^m\}_{m=1,\, m\neq n}^N) \geq (N-1)\log(B) - \mathbb{E}\mathcal{L}^n_{\text{Muscle}}(\boldsymbol{z}_i^n).$$





$$\begin{aligned} I(\boldsymbol{z}_i^1; \boldsymbol{z}_i^2, \boldsymbol{z}_i^3) &= I(\boldsymbol{z}_i^1; \boldsymbol{z}_i^2) + I(\boldsymbol{z}_i^1; \boldsymbol{z}_i^3 \mid \boldsymbol{z}_i^2) \\ &\geq 2\log(B) - \mathbb{E}\mathcal{L}^1_{\text{Muscle}}(\boldsymbol{z}_i^1) \end{aligned}$$

$$\begin{aligned} I(\boldsymbol{z}_i^1; \boldsymbol{z}_i^2) &\geq \log(B) - \mathbb{E}\mathcal{L}^{1,2}_{\text{InfoNCE}}(\boldsymbol{z}_i^1) \\ I(\boldsymbol{z}_i^1; \boldsymbol{z}_i^3) &\geq \log(B) - \mathbb{E}\mathcal{L}^{1,3}_{\text{InfoNCE}}(\boldsymbol{z}_i^1) \end{aligned}$$

Figure: Venn diagram illustrating the mutual information between model representations. Compared to pairwise alignment, the proposed Muscle loss is more effective at facilitating knowledge transfer among multiple models.

FedMuscle Methodology

Algorithm Training Procedure of FedMuscle

Input: Public dataset $\mathcal{D}$; local dataset $\mathcal{D}^n$, $n \in [N]$; number of local epochs E; number of communication rounds R; number of CL epochs T; batch size B; number of selected representation matrices M.
Output: A customized local model $\boldsymbol{\theta}^n$ for each user $n \in [N]$ on its task.
Randomly initialize $\boldsymbol{\theta}^n = \{\boldsymbol{w}^n, \boldsymbol{\phi}^n\}$ for each user $n \in [N]$.
for each communication round $r \in [R]$ **do**
    for each user $n \in [N]$ in parallel **do**
        $\{\boldsymbol{w}^n, \boldsymbol{\phi}^n\} \leftarrow$ **LocalUpdate**$(\boldsymbol{\theta}^n, \mathcal{D}^n, E)$
        for each CL epoch $t \in [T]$ **do**
            for each batch of data samples from $\mathcal{D}$ **do**
                Obtain the representation matrix $\boldsymbol{Z}^n$ and send it to the server.
                Receive $\boldsymbol{S}^n$ and $\boldsymbol{\alpha}^n$ from the server, where $\boldsymbol{S}^n$ and $\boldsymbol{\alpha}^n$ are computed based on the M representation matrices randomly selected by the server for user n.
                Update $\boldsymbol{w}^n$ by minimizing the Muscle loss.
            end for
        end for
    end for
end for

To reduce communication costs in FedMuscle, we randomly select only M of the available representation matrices from the $N-1$ users to compute $\boldsymbol{S}^n$ and $\boldsymbol{\alpha}^n$ for user n.

function LocalUpdate$(\boldsymbol{\theta}^n, \mathcal{D}^n, E)$
    for each local epoch $e \in [E]$ **do**
        Update $\boldsymbol{\theta}^n$ by minimizing the loss function $\mathcal{L}^n$ corresponding to user n's task on $\mathcal{D}^n$.
    end for
Return $\{\boldsymbol{w}^n, \boldsymbol{\phi}^n\}$

Comparison with Baselines

- FedMuscle consistently provides better overall performance for users on their respective tasks compared to the baseline algorithms.

Table: Performance of FedMuscle compared to the considered baseline algorithms.

Public Dataset	User #	Model	Task	Eval. Metric	Algorithm								
					FedMuscle (Ours)	SAGE	FedHeNN	CoFED	FedDF	FedRCL	SimCLR	Pseudo-labeling	Local Training
Pascal VOC	1	ViT-Base	MLC	micro-F1	46.33$_{\pm 0.12}$	41.97$_{\pm 0.34}$	41.27$_{\pm 0.48}$	47.47$_{\pm 0.12}$	42.43$_{\pm 0.17}$	41.77$_{\pm 0.34}$	40.80$_{\pm 0.45}$	45.93$_{\pm 0.52}$	42.17$_{\pm 0.24}$
	2	ViT-Small	MLC	micro-F1	49.77$_{\pm 0.29}$	45.37$_{\pm 0.59}$	45.87$_{\pm 0.46}$	48.80$_{\pm 0.36}$	43.67$_{\pm 1.29}$	44.07$_{\pm 0.29}$	44.83$_{\pm 0.46}$	48.07$_{\pm 0.45}$	43.67$_{\pm 0.59}$
	3	ViT-Large	MLC	micro-F1	49.40$_{\pm 0.50}$	44.10$_{\pm 0.08}$	48.23$_{\pm 0.66}$	49.77$_{\pm 0.49}$	42.57$_{\pm 0.19}$	42.43$_{\pm 0.42}$	44.53$_{\pm 0.68}$	48.20$_{\pm 0.28}$	42.93$_{\pm 0.41}$
	4	ViT-Base	IC100	Accuracy	36.67$_{\pm 0.34}$	24.50$_{\pm 0.57}$	24.10$_{\pm 0.51}$	24.67$_{\pm 0.29}$	23.93$_{\pm 0.41}$	25.27$_{\pm 0.09}$	27.43$_{\pm 0.42}$	21.40$_{\pm 0.33}$	24.77$_{\pm 0.42}$
	5	ViT-Small	IC100	Accuracy	29.93$_{\pm 0.54}$	25.13$_{\pm 0.47}$	24.83$_{\pm 1.10}$	23.70$_{\pm 1.31}$	24.70$_{\pm 0.45}$	27.23$_{\pm 0.61}$	23.60$_{\pm 0.16}$	22.77$_{\pm 1.10}$	24.70$_{\pm 0.36}$
	6	ViT-Tiny	IC10	Accuracy	66.57$_{\pm 1.01}$	43.33$_{\pm 1.15}$	41.63$_{\pm 1.33}$	43.40$_{\pm 0.37}$	43.20$_{\pm 0.24}$	44.63$_{\pm 0.12}$	49.03$_{\pm 0.80}$	43.77$_{\pm 1.73}$	43.77$_{\pm 0.62}$
	Δ (%) $\uparrow$				**+26.70**	+0.96	-0.41	+5.83	-0.82	+2.17	+3.57	+1.64	0.00
COCO	1	ViT-Base	MLC	micro-F1	49.10$_{\pm 0.45}$	41.97$_{\pm 0.33}$	42.07$_{\pm 0.66}$	50.87$_{\pm 0.38}$	43.17$_{\pm 0.46}$	41.77$_{\pm 0.34}$	42.50$_{\pm 0.70}$	47.23$_{\pm 0.17}$	42.17$_{\pm 0.24}$
	2	ViT-Small	MLC	micro-F1	51.30$_{\pm 0.22}$	46.97$_{\pm 0.17}$	46.27$_{\pm 0.54}$	52.43$_{\pm 0.41}$	43.67$_{\pm 0.54}$	44.07$_{\pm 0.29}$	44.70$_{\pm 0.37}$	50.30$_{\pm 0.24}$	43.67$_{\pm 0.59}$
	3	ViT-Large	MLC	micro-F1	50.60$_{\pm 0.36}$	44.47$_{\pm 0.68}$	47.17$_{\pm 1.77}$	53.50$_{\pm 0.37}$	42.07$_{\pm 0.54}$	42.43$_{\pm 0.42}$	45.17$_{\pm 1.06}$	50.53$_{\pm 0.05}$	42.93$_{\pm 0.41}$
	4	ViT-Base	IC100	Accuracy	37.27$_{\pm 0.78}$	25.20$_{\pm 0.49}$	24.53$_{\pm 0.66}$	24.83$_{\pm 0.17}$	24.10$_{\pm 0.67}$	25.27$_{\pm 0.09}$	28.87$_{\pm 1.11}$	21.70$_{\pm 0.85}$	24.77$_{\pm 0.42}$
	5	ViT-Small	IC100	Accuracy	30.93$_{\pm 0.12}$	25.47$_{\pm 0.52}$	25.20$_{\pm 0.43}$	24.73$_{\pm 0.75}$	25.23$_{\pm 0.33}$	27.23$_{\pm 0.61}$	23.17$_{\pm 0.87}$	23.67$_{\pm 0.66}$	24.70$_{\pm 0.36}$
	6	ViT-Tiny	IC10	Accuracy	63.23$_{\pm 0.58}$	43.03$_{\pm 0.54}$	43.07$_{\pm 1.03}$	40.90$_{\pm 1.44}$	43.20$_{\pm 0.24}$	44.63$_{\pm 0.12}$	47.37$_{\pm 0.45}$	44.13$_{\pm 1.73}$	43.77$_{\pm 0.62}$
	Δ (%) $\uparrow$				**+28.65**	+2.31	+2.57	+9.85	-0.25	+2.17	+4.49	+4.86	0.00

Impact of Muscle Loss in FedMuscle

- Muscle loss outperforms the Gramian-based contrastive loss (ICLR 2025), increasing Δ by 11.2%, 28.4%, and 11.1% on the Pascal VOC, COCO, and CIFAR-100 datasets, respectively.
- These improvements stem from the ability of Muscle loss to more effectively capture dependencies among the representations obtained from multiple models.

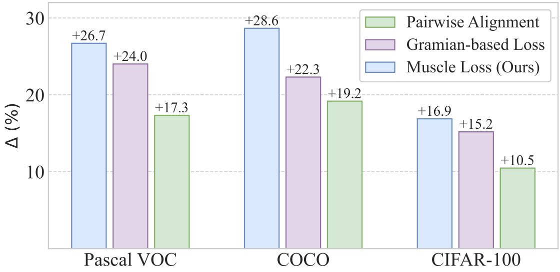


Figure: Performance of FedMuscle using the proposed Muscle loss function, compared to the Gramian-based contrastive loss and pairwise alignment.

Ablation Studies

- A greater number of communication rounds leads to improved performance for FedMuscle.
- FedMuscle is flexible enough to achieve high performance in terms of Δ while maintaining a low communication cost by appropriately adjusting R, T, and $|\mathcal{D}|$.

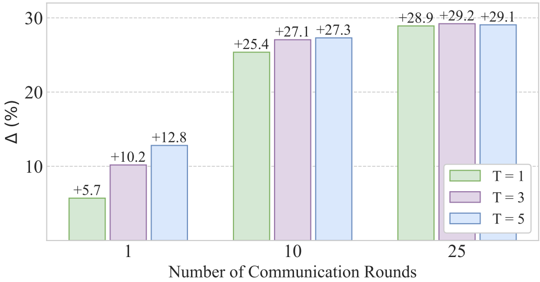


Figure: Impact of the number of local epochs E, communication rounds R, and CL epochs T on FedMuscle performance.

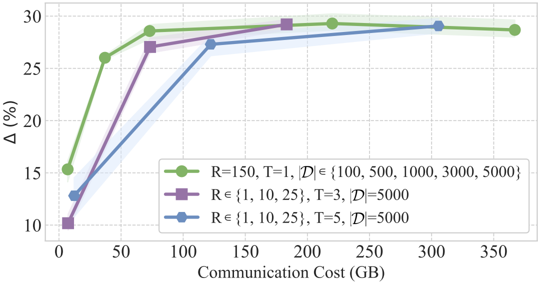


Figure: Impact of public dataset size $|\mathcal{D}|$ and the number of communication rounds R on FedMuscle's communication cost.

Conclusion

- To address both model and task heterogeneity in FMTL settings, we proposed learning a shared representation space across tasks rather than exchanging model parameters.
- We proposed Muscle loss, a novel CL objective that aligns representations from multiple models simultaneously by capturing dependencies among all models' representations.
- We developed FedMuscle, a practical FMTL algorithm where users transmit representations obtained from a public dataset to a server, which then computes aggregated matrices and weighting coefficients.
- Future work will investigate how Muscle loss enhances performance in multiview and multimodal representation learning settings.