Towards Memory-Efficient Foundation Models in Medical Imaging: A Federated Learning and Knowledge Distillation Approach

Afsaneh Mahanipour, Abdullah-Al-Zubaer Imran, Hana Khamfroush
Department of Computer Science
University of Kentucky
Lexington, KY 40506
ama654@uky.edu, aimran@uky.edu, khamfroush@cs.uky.edu

Abstract

The rapid development of medical foundation models has shown great promise for various healthcare applications. However, fine-tuning these models for downstream tasks remains challenging due to privacy concerns that limit centralized data collection from diverse sources. Federated learning (FL) offers a privacy-preserving solution by enabling multiple clients to collaboratively train a global model without sharing their local data. Despite its advantages, FL must balance model performance with communication and computation costs. Existing approaches often use parameter-efficient fine-tuning (PEFT) techniques to reduce communication overhead by transmitting fewer parameters. However, these methods require clients to host large foundation models, which is impractical for clients with limited memory. Meanwhile, conventional knowledge distillation (KD) methods fall short in FL due to misalignment between pre-trained foundation models and specific downstream tasks. To overcome these limitations, we propose Federated Reprogramming Knowledge Distillation (FedRD), a method that uses lightweight student models in clients and a medical foundation model on the server. A reprogramming module aligns the foundation model's feature space with the downstream task, enabling student models to mimic this representation collaboratively. FedRD significantly reduces memory and computation requirements while maintaining high accuracy. Experiments on three medical imaging datasets under non-IID data distributions demonstrate that FedRD outperforms federated KD and PEFT methods, offering an effective trade-off between accuracy, communication, and computational efficiency.

1 Introduction

Large-scale pre-trained foundation models are rapidly being developed for a wide range of downstream tasks Bommasani et al. (2021); Abukadah et al. (2024); Zhou et al. (2024a); Munia and Imran (2025). However, developing medical foundation models remains challenging due to the limited availability of labeled data Zhang and Metaxas (2024). In many cases, publicly available datasets have already been used, making it necessary to rely on private or protected data to improve model generalization. Unfortunately, individual healthcare institutions often lack enough data for specific tasks, and combining data across centers is generally not feasible. This challenge is primarily due to strict data privacy regulations, such as the EU's GDPR, Singapore's PDPA, and China's cybersecurity laws, which prohibit sharing raw patient data for centralized model fine-tuning Chen et al. (2024).

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: The Second Workshop on GenAI for Health: Potential, Trust, and Policy Compliance.

This problem can be addressed by Federated Learning (FL), a decentralized machine learning approach that enables clinics with varying resources and heterogeneous data to collaboratively train a global model without sharing raw data McMahan et al. (2017); Mahanipour and Khamfroush (2025). However, integrating FL with large foundation models is often impractical due to the substantial computational and communication overhead involved in optimizing and transmitting billions of parameters between clients and the server Wu et al. (2024). To mitigate this challenge, recent studies have adopted parameter-efficient fine-tuning (PEFT) methods, such as adapters Xin et al. (2024); Lu et al. (2023); He et al. (2023) and Low-Rank Adaptation (LoRA) Hu et al. (2022), which allow fine-tuning and exchanging only a small subset of model parameters. While these methods significantly reduce training and communication costs, they do not resolve memory and storage constraints, as the full foundation model still needs to reside on each client device.

Another approach is knowledge distillation (KD) Hinton et al. (2015); Liu et al. (2023), a model compression and enhancement technique that transfers knowledge from a foundation model to a smaller model, treating the foundation model as the teacher and the smaller model as the student. However, the effectiveness of KD may be limited by a lack of alignment between the pre-trained foundation model and the student model, particularly when the foundation model's pre-training data is inconsistent with the specific downstream task. Model reprogramming is one method that can mitigate this problem Xu et al. (2023); Zhou et al. (2024b).

In this work, we introduce the first federated reprogramming knowledge distillation framework, designed to enable the use of medical foundation models for downstream tasks in a distributed setting. In our approach, a frozen foundation model is hosted on the server, while lightweight student models are deployed on clients. Unlike federated PEFT methods, our framework does not require a foundation model on each client, eliminating the memory and design complexity associated with adapting large models to resource-constrained devices. To improve task relevance, a reprogramming module is incorporated on the server to align the foundation model's feature space with the target downstream task. These reprogrammed features are then used to guide the training of student models, allowing clients to learn more effective decision boundaries through distillation. Our key contributions are summarized as follows:

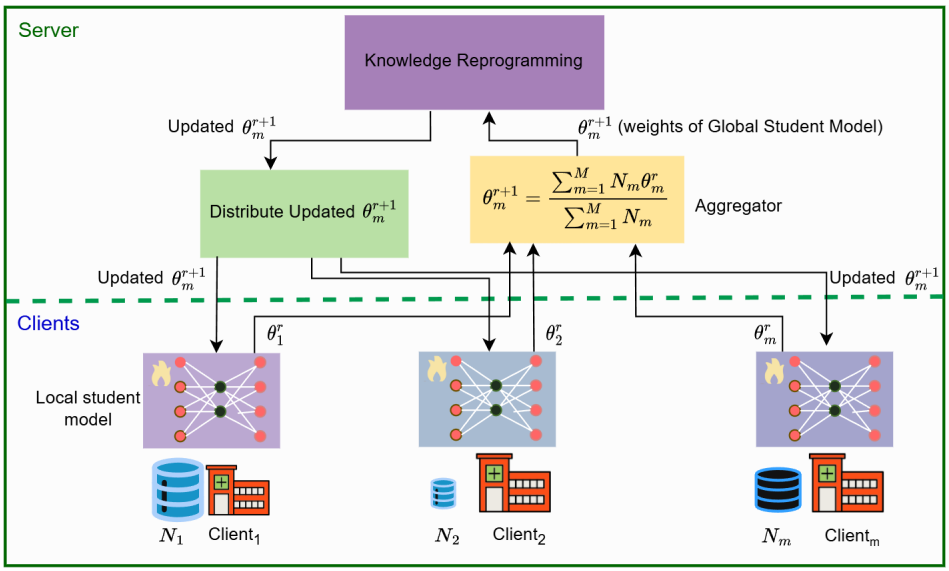
1. We propose the first federated reprogramming knowledge distillation (FedRD) method to adapt medical foundation models for downstream tasks in distributed environments. FedRD enables the training of lightweight student models on clients by transferring reprogrammed knowledge in a more communication- and computation-efficient manner.

2. We conduct extensive experiments on three datasets from different downstream tasks. The results show that our approach achieves a better balance between model accuracy, communication overhead, and computational cost compared to existing federated PEFT and KD methods.

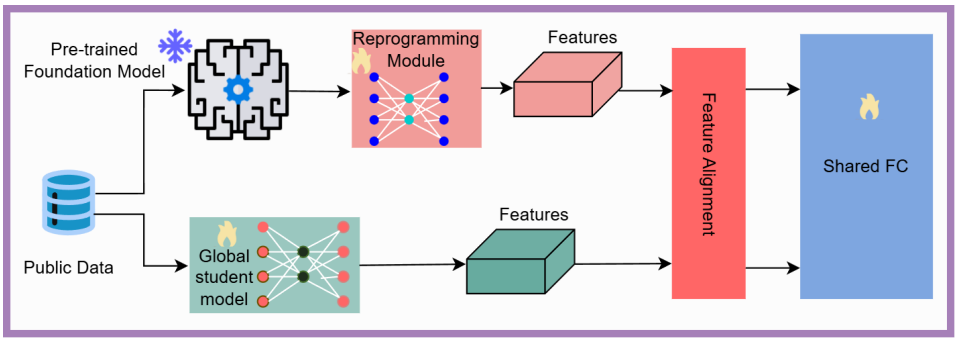
2 Federated Reprogramming Knowledge Distillation

2.1 Problem Statement

We consider a two-tier federated learning architecture. The first tier comprises M clients, denoted as $\{C_1, C_2, ..., C_m, ..., C_M\}$, where each client C_m holds a local dataset $\mathcal{U}_m = \{X, Y\} = \{(x_i, y_i)\}_{i=1}^{N_m}$, consisting of N_m data instances. The number of instances may vary across clients, reflecting a non-uniform data distribution. The second tier consists of a central server, denoted as s, which coordinates the training process. To preserve the federated nature of the system, M must be at least two, as a single client would reduce the setup to a centralized scenario. A pre-trained medical foundation model, denoted by F_t, resides on the server and acts as the teacher model. In parallel, a lightweight student model, denoted by F_s^{global}, is initialized and maintained on the server. At each communication round, this student model is broadcast to all clients, where it is referred to locally as F_s^{local}. The goal is to transfer knowledge from the foundation model to the student models in a way that maintains high performance while reducing communication and computation costs across the system.



((a)) Overview of the proposed FedRD framework.



((b)) Server-side Knowledge Reprogramming block.

Figure 1: Illustration of the proposed FedRD framework. (a) shows the overall structure of the method, and (b) details the server-side Knowledge Reprogramming block.

2.2 Proposed Method

In this section, we introduce our proposed method, federated reprogramming knowledge distillation (FedRD). The goal is to collaboratively train lightweight student models on clients by leveraging the knowledge of a pre-trained teacher foundation model, without deploying the large foundation model on resource-constrained clients. This design choice addresses the practical limitations of memory, computation, and communication on the client side. FedRD utilizes a model reprogramming strategy on the server to adapt the foundation model to the specific downstream task, ensuring that the extracted features are both consistent and task-relevant before knowledge distillation occurs. Rather than retraining the entire foundation model, model reprogramming Xu et al. (2023); Zhou et al. (2024b) enables efficient cross-domain adaptation by introducing lightweight trainable components including input transformation layers and an output mapping layer. This significantly reduces the computational overhead while taking advantage of the foundation model's rich representational power.

Before initiating server-client communication, the pre-trained teacher foundation model is deployed on the server. At the start of the training process (round $r = 1$), the server initializes a randomly configured lightweight student model, referred to as the global student model, and distributes it to all clients, denoted as θ_m^r. Each client then performs local training to optimize its local student model by minimizing the following loss function L_m^r:

$$L_m^r(\theta_m^r) = \frac{1}{N_m} \sum_{i=1}^{N_m} \mathcal{L}(\theta_m^r(x_i), y_i) \tag{1}$$

where $\mathcal{L}$ denotes the training loss function, such as cross-entropy for standard classification tasks, and x_i and y_i represent the local input data and corresponding labels. After completing local training, each client sends its updated student model to the server. The server then aggregates these local models using a weighted average to update the global student model:

$$\theta_m^{r+1} = \frac{\sum_{m=1}^{M} N_m \theta_m^r}{\sum_{m=1}^{M} N_m} \tag{2}$$

On the server side, to leverage the pre-trained teacher foundation model, a trainable reprogramming module is employed as shown in Fig. 1(b). This module consists of standard residual blocks $\phi(.)$ as input transformation layers and a fully connected (FC) layer $g(.)$ as the output mapping layer. A publicly available dataset related to the downstream task is then used to jointly train both the reprogramming module and the global student model through a co-training mechanism. This setup ensures that the reprogrammed features extracted from the foundation model can be effectively mimicked by the student model's features, allowing the student to learn decision boundaries that closely resemble those of the teacher model.

To further enhance feature alignment and enable robust knowledge distillation, Centered Kernel Alignment (CKA) Kornblith et al. (2019) is used to measure the similarity between the reprogrammed features of the foundation model and those extracted by the student model. The overall training loss is thus formulated as follows:

$$\mathcal{L}_{train} = \mathcal{L}_{CE}(y, z_s) + \alpha \mathcal{L}_{CE}(y, z_t) + \beta(\mathcal{L}_{KL}(z_t, z_s) + \mathcal{L}_{CKA}(f_t, f_s)) \tag{3}$$

where $z_t = g(\phi(F_t(x)))$ and $z_s = g(F_s(x))$ represent the output logits of the teacher foundation model and the global student model, respectively. The functions $\phi(.)$ and $F_t(.)$ denote the input reprogramming module and the frozen teacher model, while $F_s(.)$ is the student model and $g(.)$ is the shared FC classifier. The hyperparameters α and β control the contributions of different loss components. In addition to the standard cross-entropy (CE) loss, the Kullback-Leibler (KL) divergence is used as a logits-based knowledge distillation loss. Furthermore, the CKA-based feature alignment loss ($\mathcal{L}_{CKA}$) is computed as follows:

$$\mathcal{L}_{CKA}(f_t, f_s) = -\frac{HSIC(P, Q)}{\sqrt{HSIC(P, P).HSIC(Q, Q)}} \tag{4}$$

where f_t and f_s denote the reprogrammed features extracted from the foundation model and the features extracted from the student model, respectively. The pairwise feature similarity matrices are defined as $P = f_t f_t^\top$ and $Q = f_s f_s^\top$. To measure the similarity between these feature representations, we compute the HSIC Criterion as:

$$HSIC(P, Q) = \frac{P'.Q'}{(n-1)^2} \tag{5}$$

where $P' = HPH$, $Q' = HQH$, and $H = I_n - \frac{1}{n}\mathbf{1}\mathbf{1}^\top$ is the centering matrix, with n representing the batch size.

After updating the global student model on the server, it is redistributed to all clients for the next training round as shown in Fig. 1(a).

3 Experiments

Datasets: The proposed method is evaluated on three publicly available medical image datasets, each representing a different downstream task: Melanoma Tschandl et al. (2018), COVID-19 (including two datasets from Rahimzadeh et al. (2021); Yang et al. (2020)), and Brain Tumor Saleh et al. (2020). Detailed information about each dataset is provided in Table 1. For datasets without official train-test splits, we used an 80-20 division for training and testing data.

Table 1: Characteristics of downstream datasets for different tasks.

Dataset	Task	Modality	Classes	Data Size
ISIC2018 Tschandl et al. (2018)	Melanoma	RGB	7	11527
COVID Rahimzadeh et al. (2021); Yang et al. (2020)	COVID-19	CT	2	13716
BTC Saleh et al. (2020)	Brain tumor	MRI	4	3264

Teacher Foundation Models and Lightweight Student Models: To evaluate the proposed method, we employ two medical foundation models with distinct training approaches: PMC-CLIP Lin et al. (2023) and LVM-Med MH Nguyen et al. (2023). PMC-CLIP is trained using contrastive learning on 1.6 million image-caption pairs and utilizes a ResNet-50 architecture as its visual encoder. In contrast, LVM-Med is developed through self-supervised learning on 1.3 million medical images and is based on the ViT-B (Vision Transformer Base) architecture. In all experiments, the parameters of both foundation models are kept frozen. Additionally, ResNet-18, ShuffleNet, and MobileNet are used as lightweight student models on the client side.

Implementation settings: We conduct experiments in a simulated federated learning environment with three clients, selected through trial and error. Each client holds a non-IID partition of the dataset created according to the Dirichlet distribution with a concentration parameter of 0.5. Model training is performed using the AdamW optimizer with a learning rate of $5e-3$. The loss function includes two hyperparameters, α and β, which are both initialized to 1 and linearly decreased throughout the training process Zhou et al. (2024b). Each experiment runs for 50 training rounds, and the results are reported as the average classification accuracy over three independent runs. All experiments are conducted using PyTorch 2.5.1 and Tesla V100-SXM2-32GB GPU.

Results and Analysis: For comparison, we select several widely used KD methods: Hint Romero et al. (2014), VID Ahn et al. (2019), SemCKD Chen et al. (2021), and Crd Tian et al. (2019), and adapt them from centralized training to the federated learning setting using the same procedure as vanilla FL, while applying their respective loss functions. Additionally, we include a centralized reprogramming distillation (Cntr-RD) Zhou et al. (2024b) baseline to provide a more comprehensive evaluation. Table 2 summarizes the accuracy performance of all methods across different datasets, teacher foundation models, and student models. In addition, Fig. 2 provides a comparison between the proposed method and other KD approaches on the COVID and ISIC datasets, using PMC-CLIP as the teacher model and ResNet-18 as the student model, evaluated in terms of F1 score. The results demonstrate that our proposed method consistently outperforms the federated KD baselines. This improvement highlights the benefit of aligning the feature space of the foundation model with the downstream task, which enhances the quality of knowledge transfer compared to direct federated distillation approaches. As a result, our method achieves superior performance without incurring additional computation or communication costs.

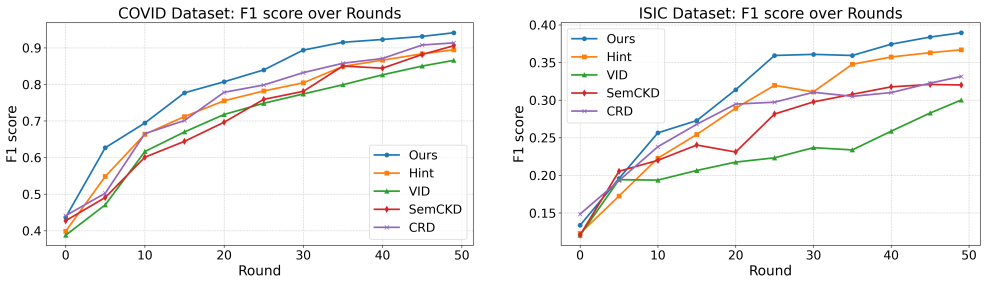
Table 3 compares our proposed method with PEFT approaches, specifically Adapter Lu et al. (2023) and LoRA Hu et al. (2022), which we adapt from centralized to federated settings. In these PEFT methods, each client is required to host a full foundation model, resulting in high memory consumption, as reflected in the parameter size column. In contrast, our method significantly reduces both computational and memory requirements, as shown by the lower GPU utilization and smaller parameter size. Here, PMC-CLIP is used as the foundation model in all three methods, while ResNet-18 serves as the student model in our proposed method. Additionally, it achieves higher accuracy while maintaining a reasonable communication cost. Overall, the results demonstrate that our reprogramming-based knowledge distillation method offers a better trade-off between performance, computation/communication efficiency, and training time in federated learning environments.

4 Conclusion

In this work, we propose Federated Reprogramming Knowledge Distillation (FedRD), a novel approach that differs from existing federated parameter-efficient fine-tuning (PEFT) and knowledge distillation (KD) methods. Instead of requiring each client to host a full foundation model, FedRD

Table 2: Comparison of the proposed method with state-of-the-art knowledge distillation methods in terms of accuracy.

Dataset	Student	Teacher	Cntr-RD	Hint	VID	SemCKD	Crd	Ours
COVID	ResNet18	PMC-CLIP	0.9526	0.9215	0.9165	0.9319	0.9372	**0.9587**
		LVM-Med	**0.9552**	0.9345	0.9142	0.9449	0.8969	0.9484
	ShuffleNet	PMC-CLIP	0.7774	0.8242	0.8450	0.8382	0.8423	**0.8529**
		LVM-Med	0.7786	0.8546	0.8757	0.8619	0.8624	**0.8851**
	MobileNet	PMC-CLIP	0.9011	0.8958	0.8715	0.8669	0.9050	**0.9685**
		LVM-Med	0.8730	0.9176	0.9137	0.8883	0.8875	**0.9674**
ISIC	ResNet18	PMC-CLIP	0.7156	0.6994	0.6687	0.6797	0.6878	**0.7169**
		LVM-Med	0.7235	0.6943	0.6753	0.6891	0.6931	**0.7282**
	ShuffleNet	PMC-CLIP	0.6779	0.6736	0.6545	0.6604	0.6534	**0.7030**
		LVM-Med	0.6376	0.6839	0.6563	0.6697	0.6481	**0.7037**
	MobileNet	PMC-CLIP	0.6693	0.6658	0.6473	0.6666	0.6515	**0.6971**
		LVM-Med	0.6647	0.6684	0.6632	0.6521	0.6554	**0.6825**
BTC	ResNet18	PMC-CLIP	0.2944	0.2919	0.2855	0.2944	0.2923	**0.2970**
		LVM-Med	0.2969	0.2893	0.2718	0.2867	0.2784	**0.2995**
	ShuffleNet	PMC-CLIP	0.2741	0.2779	0.2858	0.2792	0.2804	**0.2978**
		LVM-Med	0.2791	0.2843	0.2868	0.2893	0.2886	**0.2944**
	MobileNet	PMC-CLIP	0.2740	0.2817	0.2861	0.2859	0.2833	**0.2998**
		LVM-Med	0.2706	0.2766	0.2953	0.2937	0.2867	**0.2969**



((a)) COVID dataset ((b)) ISIC dataset

Figure 2: Comparison of the proposed method with KD methods using PMC-CLIP and ResNet18 in terms of F1 score.

collaboratively trains a reprogramming module alongside lightweight student models to adapt a pre-trained medical foundation model for downstream image classification tasks in a federated setting. We conduct extensive experiments across three diverse medical imaging datasets using two medical foundation models and three lightweight student architectures. The results show that FedRD achieves a strong balance between model performance, communication efficiency, and computational cost, outperforming existing federated PEFT and KD baselines.

Acknowledgement. This work is funded by career grant provided by the National Science Foundation (NSF) under the grant number 2340075.

Table 3: Comparison of the proposed method with PEFT methods in terms of parameter size, communication cost, GPU utilization, and training time.

PMC-CLIP						
Dataset	Method	Acc.	Param Size (MB)	Comm. (MB/round)	GPU Util. (MB)	Time(s)
COVID	Adapter	0.7773	1297.42	48.06	100	279.69
	LoRA	0.6175	581.46	0.76	57.71	83.13
	Ours	0.9587	42.67	256.29	70.79	156
ISIC	Adapter	0.1466	337.50	27.04	90.41	44.37
	LoRA	0.0324	581.46	0.76	86.1	64
	Ours	0.7169	42.68	256.35	42.8	200.79
BTC	Adapter	0.1878	1297.42	48.06	93.24	47.49
	LoRA	0.2435	581.46	0.76	87.48	13.43
	Ours	0.2970	42.67	256.32	59.18	46.80

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

R. Bommasani, D. A. Hudson, E. Adeli, R. Altman, S. Arora, S. von Arx, M. S. Bernstein, J. Bohg, A. Bosselut, E. Brunskill *et al.*, "On the opportunities and risks of foundation models," *arXiv preprint arXiv:2108.07258*, 2021.

H. Abukadah, M. Fereidouni, and A. Siddique, "Mapping natural language intents to user interfaces through vision-language models," in *2024 IEEE 18th International Conference on Semantic Computing (ICSC)*. IEEE, 2024, pp. 237–244.

Y. Zhou, Z. Zhao, H. Li, S. Du, J. Yao, Y. Zhang, and Y. Wang, "Exploring training on heterogeneous data with mixture of low-rank adapters," *arXiv preprint arXiv:2406.09679*, 2024.

N. Munia and A. A. Z. Imran, "Prompting medical vision-language models to mitigate diagnosis bias by generating realistic dermoscopic images," in *2025 IEEE 22nd International Symposium on Biomedical Imaging (ISBI)*. IEEE, 2025, pp. 1–4.

S. Zhang and D. Metaxas, "On the challenges and perspectives of foundation models for medical image analysis," *Medical image analysis*, vol. 91, p. 102996, 2024.

H. Chen, Y. Zhang, D. Krompass, J. Gu, and V. Tresp, "Feddat: An approach for foundation model finetuning in multi-modal heterogeneous federated learning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 10, 2024, pp. 11 285–11 293.

B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Artificial intelligence and statistics*. PMLR, 2017, pp. 1273–1282.

A. Mahanipour and H. Khamfroush, "Embedded federated feature selection with dynamic sparse training: Balancing accuracy-cost tradeoffs," *arXiv preprint arXiv:2504.05245*, 2025.

Y. Wu, C. Desrosiers, and A. Chaddad, "Facmic: Federated adaptative clip model for medical image classification," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2024, pp. 531–541.

Y. Xin, J. Yang, S. Luo, H. Zhou, J. Du, X. Liu, Y. Fan, Q. Li, and Y. Du, "Parameter-efficient fine-tuning for pre-trained vision models: A survey," *arXiv preprint arXiv:2402.02242*, 2024.

W. Lu, X. Hu, J. Wang, and X. Xie, "Fedclip: Fast generalization and personalization for clip in federated learning," *arXiv preprint arXiv:2302.13485*, 2023.

X. He, C. Li, P. Zhang, J. Yang, and X. E. Wang, "Parameter-efficient model adaptation for vision transformers," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 37, no. 1, 2023, pp. 817–825.

E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, W. Chen *et al.*, "Lora: Low-rank adaptation of large language models." *ICLR*, vol. 1, no. 2, p. 3, 2022.

G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531*, 2015.

X. Liu, L. Li, C. Li, and A. Yao, "Norm: Knowledge distillation via n-to-one representation matching," *arXiv preprint arXiv:2305.13803*, 2023.

S. Xu, J. Yao, R. Luo, S. Zhang, Z. Lian, M. Tan, B. Han, and Y. Wang, "Towards efficient task-driven model reprogramming with foundation models," *arXiv preprint arXiv:2304.02263*, 2023.

Y. Zhou, S. Du, H. Li, J. Yao, Y. Zhang, and Y. Wang, "Reprogramming distillation for medical foundation models," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2024, pp. 533–543.

S. Kornblith, M. Norouzi, H. Lee, and G. Hinton, "Similarity of neural network representations revisited," in *International conference on machine learning*. PMLR, 2019, pp. 3519–3529.

P. Tschandl, C. Rosendahl, and H. Kittler, "The ham10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions," *Scientific data*, vol. 5, no. 1, pp. 1–9, 2018.

M. Rahimzadeh, A. Attar, and S. M. Sakhaei, "A fully automated deep learning-based network for detecting covid-19 from a new and large lung ct scan dataset," *Biomedical Signal Processing and Control*, p. 102588, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1746809421001853

X. Yang, X. He, J. Zhao, Y. Zhang, S. Zhang, and P. Xie, "Covid-ct-dataset: a ct scan dataset about covid-19," *arXiv preprint arXiv:2003.13865*, 2020.

A. Saleh, R. Sukaik, and S. S. Abu-Naser, "Brain tumor classification using deep learning," in *2020 International Conference on Assistive and Rehabilitation Technologies (iCareTech)*. IEEE, 2020, pp. 131–136.

W. Lin, Z. Zhao, X. Zhang, C. Wu, Y. Zhang, Y. Wang, and W. Xie, "Pmc-clip: Contrastive language-image pre-training using biomedical documents," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2023, pp. 525–536.

D. MH Nguyen, H. Nguyen, N. Diep, T. N. Pham, T. Cao, B. Nguyen, P. Swoboda, N. Ho, S. Albarqouni, P. Xie *et al.*, "Lvm-med: Learning large-scale self-supervised vision models for medical imaging via second-order graph matching," *Advances in Neural Information Processing Systems*, vol. 36, pp. 27 922–27 950, 2023.

A. Romero, N. Ballas, S. E. Kahou, A. Chassang, C. Gatta, and Y. Bengio, "Fitnets: Hints for thin deep nets," *arXiv preprint arXiv:1412.6550*, 2014.

S. Ahn, S. X. Hu, A. Damianou, N. D. Lawrence, and Z. Dai, "Variational information distillation for knowledge transfer," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 9163–9171.

D. Chen, J.-P. Mei, Y. Zhang, C. Wang, Z. Wang, Y. Feng, and C. Chen, "Cross-layer distillation with semantic calibration," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 35, no. 8, 2021, pp. 7028–7036.

Y. Tian, D. Krishnan, and P. Isola, "Contrastive representation distillation," *arXiv preprint arXiv:1910.10699*, 2019.