

Reprogramming Distillation for Medical Foundation Models

Yuhang Zhou^{1,2}, Siyuan Du^{2,3}, Haolin Li^{2,3}, Jiangchao Yao^{1,2(✉)}, Ya Zhang^{1,2},
and Yanfeng Wang^{1,2}

¹ Cooperative Medianet Innovation Center, Shanghai Jiao Tong University

² Shanghai Artificial Intelligence Laboratory

³ Fudan University

{zhouyuhang, Sunarker, ya_zhang, wangyanfeng622}@sjtu.edu.cn,

{dusiyuan, lihaolin}@pjlab.org.cn

Abstract. Medical foundation models pre-trained on large-scale datasets have demonstrated powerful versatile capabilities for various tasks. However, due to the gap between pre-training tasks (or modalities) and downstream tasks (or modalities), the real-world computation and speed constraints, it might not be straightforward to apply medical foundation models in the downstream scenarios. Previous methods, such as parameter efficient fine-tuning (PEFT) methods and knowledge distillation (KD) methods, are unable to simultaneously address the task (or modality) inconsistency and achieve personalized lightweight deployment under diverse real-world demands. To address the above issues, we propose a novel framework called Reprogramming Distillation (RD). On one hand, RD reprograms the original feature space of the foundation model so that it is more relevant to downstream scenarios, aligning tasks and modalities. On the other hand, through a co-training mechanism and a shared classifier, connections are established between the reprogrammed knowledge and the knowledge of student models, ensuring that the reprogrammed feature space can be smoothly mimic by the student model of different structures. Further, to reduce the randomness under different training conditions, we design a Centered Kernel Alignment (CKA) distillation to promote robust knowledge transfer. Empirically, we show that on extensive datasets, RD consistently achieve superior performance compared with previous PEFT and KD methods. Source code is available at: <https://github.com/MediaBrain-SJTU/RD>

Keywords: Knowledge Distillation · Transfer Learning · Foundation Models · Model Reprogramming · Downstream Adaptation

1 Introduction

Large-scale pre-training has promoted the rapid development of medical foundation models, which provides powerful features for various downstream tasks [16, 18, 19, 33, 36]. However, most medical foundation models are trained in self-supervised or weakly-supervised manners by constructing surrogate tasks like

reconstruction [9], graph modeling [19] and visual-language matching [16] etc. The resulted model cannot be directly applied to specific downstream tasks, such as tumor benign-malignant diagnosis, lung disease diagnosis, etc. In addition, actual deployment scenarios may have more requirements for the model’s computing memory and inference speed [35]. For example, on some edge devices or handheld instruments from different manufacturers, it is not possible to deploy large-capacity foundation models directly. Therefore, new ways for the downstream adaptation of foundation models need to be explored.

Currently, the mainstream downstream adaptation method for foundation models is parameter efficient fine-tuning (PEFT) [29, 5, 13, 15, 12, 10], which typically introduces a small number of training parameters to achieve performance comparable to full fine-tuning. Despite its reduction in training cost, these methods might not meet the deployment constraints for memory usage and inference speed in downstream scenarios. Knowledge distillation (KD) [11, 28, 25, 21, 32, 17], as an effective means of knowledge transfer, can compress the model size without sacrificing much performance of foundation models. However, unlike the original knowledge distillation scenario, the pre-training data used by the foundation model may be task or modality inconsistent with that of the downstream task, leading to a lack of direct correlation between the foundation model and the student model, thereby weakening the gain of the foundation model for the downstream task. The close exploration [31] through model reprogramming mainly focuses on natural scenarios, while we extend this idea into the medical scenarios with the technical improvement for an end-to-end training.

Specifically, to address the main challenge, we propose an effective framework called reprogramming distillation (RD) for adapting medical foundation models on downstream tasks. Specifically, on one hand, we introduce a reprogramming module to mitigate the task (or modality) inconsistency, which helps extract features more relevant to downstream tasks. Here, we treat the foundation model as a black box with fixed parameters, which avoids complex designs of PEFT and circumvents the significant GPU costs associated with backpropagation in the large foundation model. On the other hand, in order to ensure that reprogrammed features can be smoothly mimic by downstream models, we adopt co-training mechanism to establish connections between the reprogrammed knowledge and the knowledge extracted by student models, and encourage them to learn similar decision boundaries through the shared classifier. In this process, we introduce the Centered Kernel Alignment (CKA) distillation to promote robust knowledge transfer, so as to alleviate the randomness introduced due to the various training conditions. In a nutshell, our contributions can be summarized:

- We design a new framework called Reprogramming Distillation (RD) to facilitate downstream adaptation of foundation models. RD can align inconsistencies of tasks and modalities, achieve personalized lightweight under different deployment conditions and transfer downstream knowledge more effectively, which cannot be achieved by ordinary PEFT and KD methods.
- We propose two core components for RD, namely, the co-training reprogramming and the CKA distillation. The former is used to align tasks and

modalities, and make the reprogrammed feature space easier mimic by downstream models; The latter is employed for robust feature transfer to reduce the randomness under different training conditions.

- We select three different types of medical foundation models to conduct extensive experiments on five datasets with different tasks and modalities. The results indicate that RD consistently improves performance under different training conditions and significantly outperforms previous PEFT and KD methods, especially for downstream scenarios with small data size.

2 Reprogramming distillation

2.1 Preliminary

Let F_t represents the medical foundation model pre-trained on broad data, namely, the teacher model, and F_s represents the lightweight model to be deployed, namely, the student model. Given the downstream data $D = \{x_i, y_i\}_{i=1}^N$, where N is the training data size, our goal is to transfer the knowledge relevant to downstream tasks from F_t to F_s , so that F_s can not only leverage the knowledge in foundation models but also ensure efficiency in memory and computation for inference. In the following, we present the two components of RD: the co-training reprogramming and centered kernel alignment distillation.

2.2 Co-training Reprogramming

In order to alleviate the inconsistency of tasks or/and modalities between pre-training data and downstream data, we propose to reprogram the feature space of the foundation model to more effectively extract knowledge relevant to downstream tasks from the foundation model. Basically, model reprogramming [4, 31] is a technique that involves reusing pre-trained models developed in the source domain and training only the added input transformation layers and output mapping layer to solve tasks in the target domain, thereby achieving resource-efficient cross-domain adaptation. Inspired by this spirit, we introduce a trainable reprogramming module which includes the input transformation layers, i.e. $\phi(\cdot)$, and the output mapping layer, i.e., the classifier $g(\cdot)$, where $\phi(\cdot)$ can be any model structure and $g(\cdot)$ is a fully connected layer. Here, to ensure the simplicity and generality of the method, we use standard residual blocks for $\phi(\cdot)$.

Although standard model reprogramming has been proven effective from both theoretical and experimental perspectives [4], our goal is not only to adapt to downstream tasks but also to enable the adapted model to be deployed in a personalized and lightweight manner according to downstream medical scenarios. Considering that different model structures may have different tendencies in feature extraction, to ensure that the reprogrammed features can be smoothly mimic by the student model, we propose to use the co-training mechanism. Specifically, we propose pre-training F_s concurrently with training $\phi(\cdot)$ and sharing the classifier $g(\cdot)$ to promote alignment in decision boundaries between F_s

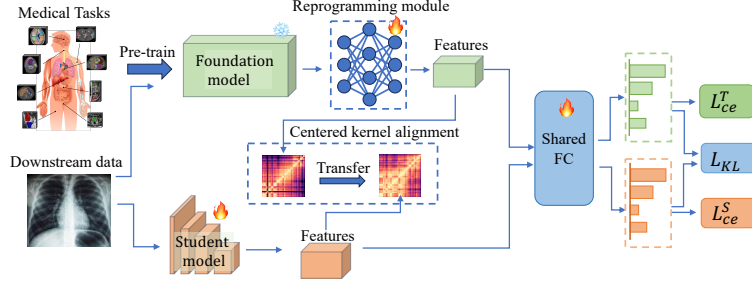


Fig. 1: The overview of RD. During training, only the foundation model is fixed.

and F_t . This approach acts as a conduit to enhance the relationship between the reprogrammed features and the features extracted by the student model. The corresponding parts are illustrated in Figure 1.

In addition to using the standard supervised loss such as the cross-entropy loss to learn downstream tasks, we also employ logits knowledge distillation [11] to enhance the relationship between the reprogrammed features and the features extracted by the student model. Specifically, we denote the logits output by the foundation model and the student model as z_t and z_s respectively, then the training loss can be written as

$$\mathcal{L}_{train} = \mathcal{L}_{CE}(y, z_s) + \alpha \mathcal{L}_{CE}(y, z_t) + \beta \mathcal{L}_{KL}(z_t, z_s), \quad (1)$$

where $z_s = g(F_s(x))$, $z_t = g(\phi(F_t(x)))$, α and β are hyper-parameters and KL is the Kullback-Leibler divergence.

2.3 Centered Kernel Alignment Distillation

During downstream adaptation, various training conditions such as different model structures, data distributions, random seeds, etc., may introduce more unnecessary noise and increase uncertainty in feature distillation [8]. To achieve robust feature distillation, we suggest using Central Kernel Alignment (CKA) distillation. CKA focuses on the correlation and higher-order information in features, which can reduce training instability, and is commonly used to identify corresponding layers between networks trained with different initializations [14].

Specifically, we denote the extracted features before the shared classifier from the foundation model and the student model as f_t and f_s respectively, and use $K = f_t f_t^\top$ and $L = f_s f_s^\top$ to represent the pair-wise feature similarities. Let $\mathbf{H} = \mathbf{I}_n - \frac{1}{n} \mathbf{1}\mathbf{1}^\top$ be the centering matrix and $K' = \mathbf{H}KH$, $L' = \mathbf{H}LH$, where n is the batch size. Then, the similarity of the centered similarity matrices can be measured by $\text{HSIC}(K, L) = \frac{K' \cdot L'}{(n-1)^2}$. Here, although HSIC is invariant to orthogonal transformations of features, it is not invariant to isotropic scaling [14]. To address this, we normalize HSIC and then construct the CKA loss as follows,

$$\mathcal{L}_{CKA} = - \frac{\text{HSIC}(K, L)}{\sqrt{\text{HSIC}(K, K) \cdot \text{HSIC}(L, L)}}, \quad (2)$$

Table 1: Downstream datasets with different tasks or modalities in the experiment.

Dataset	Task	Modality	Classes	Data Size
ISIC2018	Melanoma	RGB	7	11527
COVID	COVID-19	CT	2	746
BTC	Brain tumor	MRI	4	3264
BUSI	Breast cancer	Ultrasound	3	626
ChestXray	Lung diseases	X-Ray	14	7135

Table 2: Ablation experiments of different components in our method.

Di. Reprog.	Co. Reprog.	KD	CKA	Acc
				75.86
✓		✓		80.79
	✓	✓		82.27
	✓	✓	✓	85.71

which can be used to determine the correspondence between hidden layers of networks trained with different random initializations and widths, while other similarity measures cannot [20]. The final training loss can be written as

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

2.4 Discussion on advantages

Compared to previous downstream adaptation methods, our RD has the following three advantages: (1) Since the trainable parameters are completely isolated from the foundation model, there is no need to backpropagate gradients through the backbone of the foundation model during training, reducing GPU usage; (2) The medical foundation model in RD can be treated as a black box, with higher parameter privacy and broader application scenarios; (3) The model obtained after downstream adaptation is lightweight, and the structure can be customized according to scenarios, making deployment more flexible. (4) It builds a CKA to promote an end-to-end training to avoid the heuristic stage-wise training [31].

3 Experiments

3.1 Experimental Setup

Foundation models. We selected three medical foundation models trained in different ways to validate the proposed method’s effectiveness, namely PMC-CLIP [16], RadDenseNet [18], and LVM-Med [19]. PMC-CLIP is trained on 1.6 million pairs of image-caption data by contrastive learning, with a visual module structure of ResNet-50. RadDenseNet is trained on 1 million images by supervised learning, with a structure of DenseNet-121. LVM-Med is trained on 1.3 million images by self-supervised learning, with a structure of VIT-B. In our experiments, the parameters of these foundation models remain fixed.

Downstream Datasets and Lightweight Student Models. To comprehensively validate the generalization of our method, we selected five medical image datasets with various modalities and tasks as the targets for downstream adaptation, namely BUSI [2], ISIC [26], Covid [30], BTC [23] and ChestXray [6], which are summarized in Table 1. In the absence of official test sets, we divide the data as training and test sets in a 4:1 ratio. For the downstream models, we use three lightweight structures, namely ResNet18, MobileNet, and ShuffleNet.

Table 3: Comparison with KD methods. Due to space constraints, we only display all the results on the BUSI and BTC here. The complete results for the other three datasets are presented in the form of average gains in Table 5.

Dataset	Student	Teacher	VID	SemCKD	Crđ	Hint	MGD	Norm	Ours
BUSI	ResNet18 77.92	PMC-CLIP 88.31	77.92	68.83	75.32	77.92	77.27	73.38	88.31
		Densenet121 86.36	76.62	70.13	75.97	77.27	78.57	68.83	88.31
		LVM-Med 94.16	73.38	69.48	68.83	79.22	66.23	64.29	88.96
	ShuffleNet 81.82	PMC-CLIP 88.31	85.06	75.97	82.47	81.17	84.41	64.94	86.36
		Densenet121 86.36	81.82	76.62	81.17	83.77	84.41	56.49	85.71
		LVM-Med 94.16	81.82	80.52	79.87	81.17	82.47	79.87	85.71
	MobileNet 80.52	PMC-CLIP 88.31	85.06	85.06	81.82	83.12	85.06	82.47	86.36
		Densenet121 86.36	83.12	84.42	81.17	83.12	82.46	61.04	87.01
		LVM-Med 94.16	83.77	86.36	83.12	83.77	81.17	71.43	87.01
BTC	ResNet18 77.92	PMC-CLIP 80.20	77.16	78.17	68.78	77.66	75.89	77.66	80.71
		Densenet121 80.46	77.66	77.41	76.40	77.16	76.90	60.41	78.68
		LVM-Med 80.71	77.41	78.43	75.13	80.71	55.33	71.57	81.98
	ShuffleNet 78.17	PMC-CLIP 80.20	77.41	77.16	69.29	77.16	76.90	75.13	79.44
		Densenet121 80.46	77.41	77.16	76.90	75.63	76.65	55.08	79.95
		LVM-Med 80.71	76.65	78.68	75.13	77.92	79.95	67.77	80.46
	MobileNet 76.90	PMC-CLIP 80.20	77.92	77.41	61.93	76.65	77.92	74.62	79.70
		Densenet121 80.46	78.17	77.41	76.90	76.90	77.16	54.82	79.95
		LVM-Med 80.71	77.41	79.44	74.87	77.92	72.91	69.04	82.49
ISIC	ResNet18 70.50	PMC-CLIP 76.92	75.73	75.66	76.26	75.33	77.05	75.79	78.11
		Densenet121 73.41	75.99	75.99	76.32	76.52	77.18	63.76	77.18
		LVM-Med 82.74	75.93	77.18	78.77	79.23	75.99	78.37	79.76
COVID	ResNet18 75.86	PMC-CLIP 78.82	77.34	76.85	74.88	75.37	75.86	79.80	81.28
		Densenet121 77.34	77.83	77.83	73.89	77.83	79.31	73.89	78.82
		LVM-Med 83.74	76.85	80.79	75.86	79.80	73.40	82.27	82.27
ChestXray	ResNet18 92.87	PMC-CLIP 96.11	92.74	94.16	93.64	93.13	93.51	72.24	95.33
		Densenet121 94.55	93.77	92.35	94.03	93.39	92.09	90.53	94.68
		LVM-Med 94.16	93.64	93.51	93.51	94.42	92.22	81.84	96.11

Implementation details. We use AdamW optimizer with the learning rate set to $5e-3$. Each downstream dataset is trained for 240 epochs and the batch size is set to 32. The weights α and β in the loss function are initially 1 and decreases linearly with the training process. The GPUs we used are RTX 3090.

3.2 Ablation Studies

We verify the effectiveness of the different components in Table 2, where ‘‘Di. Reprog’’ and ‘‘Co. Reprog’’ represent direct reprogramming [4] and our co-training reprogramming respectively. It can be seen that replacing direct reprogramming with our co-training reprogramming results in a significant performance improvement. This indicates that the design of co-training reprogramming can better transfer knowledge to downstream models while mitigating task and modality inconsistencies. The CKA distillation also significantly improves performance, which may be attributed to robust knowledge transfer that tends to focus on the effective components in features, reducing the impact of randomness on training.

3.3 Comparisons on Downstream Tasks

Compared with KD methods. We compare the current representative KD methods, including Hint [22], AT [34], VID [1], SemCKD [3], Crđ [25], MGD [32]

Table 4: Comparison with PEFT methods. Since PEFT is mainly applied to transformer structures, the foundation model in this experiment is LVM-Med.

Dataset	Student	AdaptFormer	MeLo	VPT	VQT	LPT	Ours
BUSI	ResNet18	80.20	86.36	86.36	87.66	87.66	88.96
	ShuffleNet	78.93	83.77	84.77	84.77	84.42	85.71
	MobileNet	79.95	84.42	81.82	85.06	81.82	87.01
ISIC	ResNet18	79.76	76.65	77.84	77.12	76.79	79.76
	ShuffleNet	78.90	76.19	77.05	77.05	76.26	78.90
	MobileNet	76.79	76.98	76.52	76.46	76.92	76.79
COVID	ResNet18	80.30	75.86	76.35	76.35	80.79	82.27
	ShuffleNet	81.28	76.35	78.82	78.82	81.77	85.22
	MobileNet	78.82	77.34	77.34	79.80	75.85	84.24
BTC	ResNet18	80.20	78.43	77.92	77.92	78.93	81.98
	ShuffleNet	78.93	76.90	77.66	76.90	78.43	80.46
	MobileNet	79.95	76.65	76.65	78.93	76.40	82.49
Average		79.50	78.83	79.09	79.74	79.67	82.82

Table 5: The average gain of three foundation models on downstream datasets.

Dataset	BUSI	COVID	BTC	ChestXray	ISIC
Data Size	626	746	3264	6326	11527
PMC-CLIP	6.71	4.93	2.11	2.42	5.80
RadDensenet	5.19	6.08	1.86	0.77	5.10
LVM-Med	6.06	6.57	3.98	2.42	6.02

Table 6: Comparison on computation and speed of models using **RD** or **not**.

Cost	Params(M)	FLOPs(G)	Time(ms)
LVM-Med	41.26	65.34	77.92
ResNet18	5.33	6.77	3.13
ShuffleNet	0.60	0.55	7.59
MobileNet	1.06	1.17	5.95

and Norm [17] in the Table 3. In addition, we also report the full fine-tuning performance on the downstream dataset directly using the student model and the foundation model behind the name of model structures in the Table 3.

It can be seen that RD is superior overall to previous KD methods, even approaching the performance of fully fine-tuning foundation models, with more lightweight structures. Compared to directly training the student model, KD methods have limited improvement and obvious differences in performance gains on different datasets. This indicates that the direct knowledge distillation without eliminating inconsistencies of tasks or modalities is low-efficient, since the effectiveness of knowledge transfer may directly depend on the matching degree between the feature space of the foundation model and downstream tasks. The consistent improvement of RD on different datasets and structures demonstrates that RD is effective for downstream adaptation of the foundation model.

Compared with PEFT methods. We compare RD with PEFT methods of different types, such as adapter tuning method AdaptFormer [5], LoRA method MeLo [37], prompt tuning methods VPT [13] and LPT [7], and prefix tuning method VQT [27] in the Table 4. For a fair comparison, these methods also employ knowledge distillation to obtain the same student models. It can be seen that the performance of these PEFT methods varies insignificantly, within 1%, while RD demonstrates a very noticeable performance advantage, especially for small datasets like BUSI and COVID. As for ISIC, PEFT methods and RD are comparable in performance, indicating that the effectiveness of PEFT methods

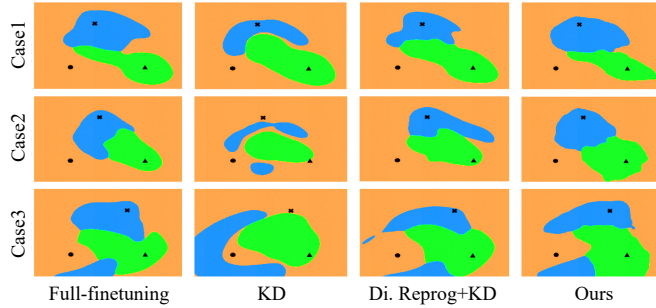


Fig. 2: Comparison of decision boundaries of different methods. Our method has decision boundaries that are more similar to the foundation model compared to other methods, which could be a reason for the better performance of ours.

relies on a sufficient amount of training data. Our method reduces the reliance on data, which is of significant importance for data-expensive medical scenarios.

3.4 Further analysis

Due to limited space, the complete results of RD on five downstream datasets, three foundation models, and three downstream models is calculated and summarized in Table 5. It can be seen that RD shows the most significant performance improvement when combined with PMC-CLIP or LVM-Med. This indicates that the gains for downstream tasks from the foundation model obtained through self-supervised learning (SSL) are greater, demonstrating stronger generalization capability of SSL. Furthermore, the performance improvement is more pronounced in downstream tasks with small data size, such as BUSI, indicating that RD can effectively reduce the demand of downstream training data.

Table 6 shows the cost comparison of deployed models using RD or not, where “Time” represents the time taken to inference each minibatch data. Combining it with Table 3, it can be seen that RD can achieve performance comparable to full-finetuning the foundation model and significantly reduce the computational cost during inference, improving the efficiency of the deployed model.

We also compared the decision boundaries of the trained models following [24], as shown in Figure 2. It can be seen that ours has decision boundaries more similar to those of fully fine-tuned foundation models compared to other baselines. This indicates that RD can better transfer knowledge from foundation models, achieving similar training effects as full-finetuning in a lightweight form.

4 Conclusion

To construct efficient and computation-friendly adaptation of the medical foundation model on the downstream tasks, it is necessary to consider the knowledge transfer under the inconsistency of data distribution and model structure.

Different from previous PEFT methods and KD methods that cannot simultaneously address the task (or modality) inconsistency and achieve personalized lightweight deployment according to different real-world demands, we propose a novel framework RD for foundation models, which can break the constraints of data distributions and model structures. Empirically, a large number of experiments on five different modal downstream tasks, three different types of models demonstrate the generalization and effectiveness of our method.

Acknowledgement. This work is supported by the National Key R&D Program of China (No. 2022ZD0160703), STCSM (No. 22511106101, No. 22511105700, No. 21DZ1100100), 111 plan (No. BP0719010) and National Natural Science Foundation of China (No. 62306178).

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

References

1. Ahn, S., Hu, S.X., Damianou, A., Lawrence, N.D., Dai, Z.: Variational information distillation for knowledge transfer. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 9163–9171 (2019)
2. Al-Dhabyani, W., Gomaa, M., Khaled, H., Fahmy, A.: Dataset of breast ultrasound images. *Data in brief* **28**, 104863 (2020)
3. Chen, D., Mei, J.P., Zhang, Y., Wang, C., Wang, Z., Feng, Y., Chen, C.: Cross-layer distillation with semantic calibration. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 35, pp. 7028–7036 (2021)
4. Chen, P.Y.: Model reprogramming: Resource-efficient cross-domain machine learning. *arXiv preprint arXiv:2202.10629* (2022)
5. Chen, S., Ge, C., Tong, Z., Wang, J., Song, Y., Wang, J., Luo, P.: Adaptformer: Adapting vision transformers for scalable visual recognition. *Advances in Neural Information Processing Systems* **35**, 16664–16678 (2022)
6. Cohen, J.P., Morrison, P., Dao, L.: Covid-19 image data collection. *arXiv preprint arXiv:2003.11597* (2020)
7. Dong, B., Zhou, P., Yan, S., Zuo, W.: Lpt: Long-tailed prompt tuning for image classification. *arXiv preprint arXiv:2210.01033* (2022)
8. Ge, L., Hu, C., et al.: Discrepancy and uncertainty aware denoising knowledge distillation for zero-shot cross-lingual named entity recognition. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 38, pp. 18056–18064 (2024)
9. He, K., Chen, X., Xie, S., Li, Y., Dollár, P., Girshick, R.: Masked autoencoders are scalable vision learners. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 16000–16009 (2022)
10. He, X., Li, C., Zhang, P., Yang, J., Wang, X.E.: Parameter-efficient model adaptation for vision transformers. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 37, pp. 817–825 (2023)
11. Hinton, G., Vinyals, O., Dean, J.: Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531* (2015)

12. Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W.: Lora: Low-rank adaptation of large language models. arXiv preprint arXiv:2106.09685 (2021)
13. Jia, M., Tang, L., Chen, B.C., Cardie, C., Belongie, S., et al.: Visual prompt tuning. In: European Conference on Computer Vision. pp. 709–727. Springer (2022)
14. Kornblith, S., Norouzi, M., Lee, H., Hinton, G.: Similarity of neural network representations revisited. In: International conference on machine learning. pp. 3519–3529. PMLR (2019)
15. Li, X.L., Liang, P.: Prefix-tuning: Optimizing continuous prompts for generation. arXiv preprint arXiv:2101.00190 (2021)
16. Lin, W., Zhao, Z., Zhang, X., Wu, C., Zhang, Y., Wang, Y., Xie, W.: Pmc-clip: Contrastive language-image pre-training using biomedical documents. arXiv preprint arXiv:2303.07240 (2023)
17. Liu, X., Li, L., Li, C., Yao, A.: Norm: Knowledge distillation via n-to-one representation matching. arXiv preprint arXiv:2305.13803 (2023)
18. Mei, X., Liu, Z., Robson, P.M., Marinelli, B., Huang, M., Doshi, A., Jacobi, A., Cao, C., et al.: Radimagenet: an open radiologic deep learning research dataset for effective transfer learning. *Radiology: Artificial Intelligence* **4**(5), e210315 (2022)
19. Nguyen, D.M., Nguyen, H., Diep, N.T., Pham, T.N., et al.: Lvm-med: Learning large-scale self-supervised vision models for medical imaging via second-order graph matching. arXiv preprint arXiv:2306.11925 (2023)
20. Nguyen, T., Raghu, M., Kornblith, S.: Do wide and deep networks learn the same things? uncovering how neural network representations vary with width and depth. arXiv preprint arXiv:2010.15327 (2020)
21. Park, W., Kim, D., Lu, Y., Cho, M.: Relational knowledge distillation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 3967–3976 (2019)
22. Romero, A., Ballas, N., Kahou, S.E., Chassang, A., Gatta, C., Bengio, Y.: Fitnets: Hints for thin deep nets. arXiv preprint arXiv:1412.6550 (2014)
23. Saleh, A., Sukaik, R., Abu-Naser, S.S.: Brain tumor classification using deep learning. In: 2020 International Conference on Assistive and Rehabilitation Technologies (iCareTech). pp. 131–136 (2020). <https://doi.org/10.1109/iCareTech49914.2020.00032>
24. Somepalli, G., Fowl, L., Bansal, A., Yeh-Chiang, P., Dar, Y., et al.: Can neural nets learn the same model twice? investigating reproducibility and double descent from the decision boundary perspective. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 13699–13708 (2022)
25. Tian, Y., Krishnan, D., Isola, P.: Contrastive representation distillation. arXiv preprint arXiv:1910.10699 (2019)
26. Tschandl, P., Rosendahl, C., Kittler, H.: The ham10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions. *Scientific data* **5**(1), 1–9 (2018)
27. Tu, C.H., Mai, Z., Chao, W.L.: Visual query tuning: Towards effective usage of intermediate representations for parameter and memory efficient transfer learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7725–7735 (2023)
28. Tung, F., Mori, G.: Similarity-preserving knowledge distillation. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 1365–1374 (2019)
29. Xin, Y., Luo, S., Zhou, H., Du, J., Liu, X., Fan, Y., Li, Q., Du, Y.: Parameter-efficient fine-tuning for pre-trained vision models: A survey. arXiv preprint arXiv:2402.02242 (2024)

30. Xingyi, Y., Xuehai, H., Jinyu, Z., Yichen, Z., et al.: Covid-ct-dataset: a ct image dataset about covid-19. arXiv preprint arXiv:2003.13865 (2020)
31. Xu, S., Yao, J., Luo, R., Zhang, S., Lian, Z., Tan, M., Han, B., Wang, Y.: Towards efficient task-driven model reprogramming with foundation models. arXiv preprint arXiv:2304.02263 (2023)
32. Yang, Z., Li, Z., Shao, M., Shi, D., Yuan, Z., Yuan, C.: Masked generative distillation. In: European Conference on Computer Vision. pp. 53–69. Springer (2022)
33. Yao, J., Zhang, S., Yao, Y., Wang, F., Ma, J., Zhang, J., Chu, Y., Ji, L., Jia, K., et al.: Edge-cloud polarization and collaboration: A comprehensive survey for ai. IEEE Transactions on Knowledge and Data Engineering **35**(7), 6866–6886 (2022)
34. Zagoruyko, S., Komodakis, N.: Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. arXiv preprint arXiv:1612.03928 (2016)
35. Zhou, Y., Li, H., Du, S., Yao, J., Zhang, Y., Wang, Y.: Low-rank knowledge decomposition for medical foundation models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11611–11620 (2024)
36. Zhou, Y., Zhao, Z., Du, S., Yao, J., Zhang, Y., Wang, Y., et al.: Exploring training on heterogeneous data with mixture of low-rank adapters. In: Forty-first International Conference on Machine Learning
37. Zhu, Y., Shen, Z., Zhao, Z., Wang, S., Wang, X., Zhao, X., Shen, D., Wang, Q.: Melo: Low-rank adaptation is better than fine-tuning for medical image diagnosis. arXiv preprint arXiv:2311.08236 (2023)