Studying distribution shifts in fully self-supervised ViTs with test-time fine-tuning

Aditya Mehrotra, Aviraj Newatia
University of Toronto

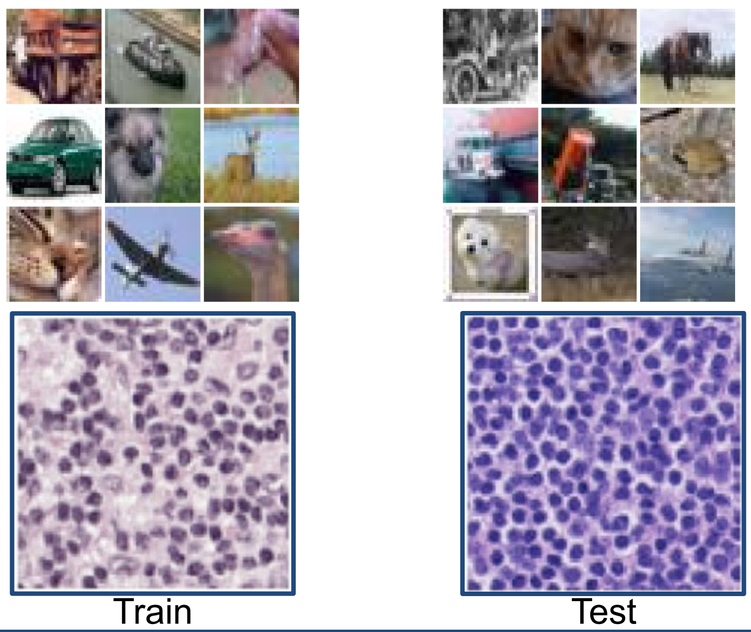
Introduction

Background: Self-Supervised Learning is a way of training robust feature-extractors in the absence of labels

Motivation: Most test-time adaptation strategies on OOD data assume that the pretrained feature extractor is tuned on an SSL + supervised objective. What if we try and do this process fully self-supervised?

Summary/Contribution: In this project, we apply common fine tuning methods from other domains (LoRA, EWC) to the DINO self-supervised training framework and observe their effects.

Datasets (ID/OOD):
- CIFAR-10/CIFAR-10.1
- CAMELYON-17 (Harmful/not harmful)



Related Work

Self-Supervised Pre Training:
- DINO [1] is a method of training vision transformers by aligning representations on same-view augmented images between a student and teacher network, where the teacher network is an EMA of the student.

Fine-tuning under domain shift:
- LoRA [2] and EWC [3] are works that allow you to finetune a network while operating with the constraint that your weights cannot move too far from initialization.

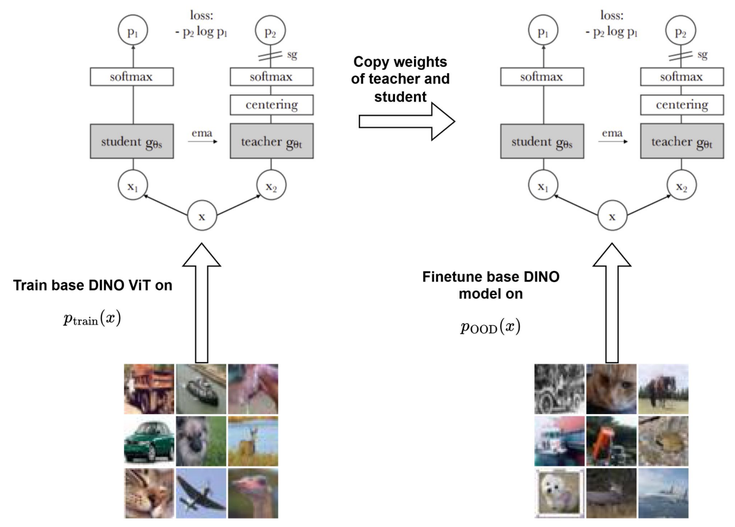
Fine-tuning with SSL objectives:
- Recent work such as TTT++ [4] form the basis of the current state of the art of self-supervised test-time-fine-tuning. This work assumes access to the training labels, which isn't always feasible in some settings (Ex: gigapixel imaging).

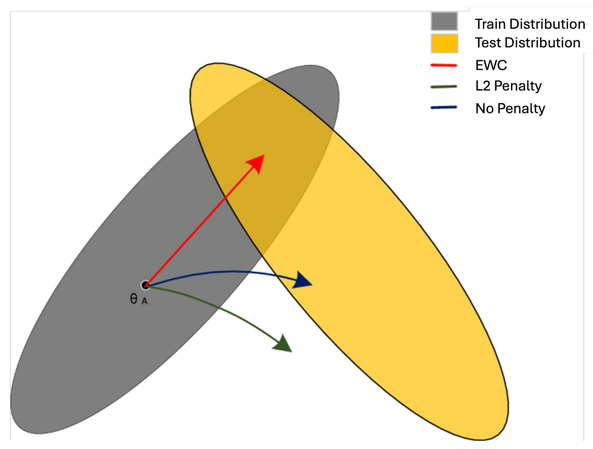
References

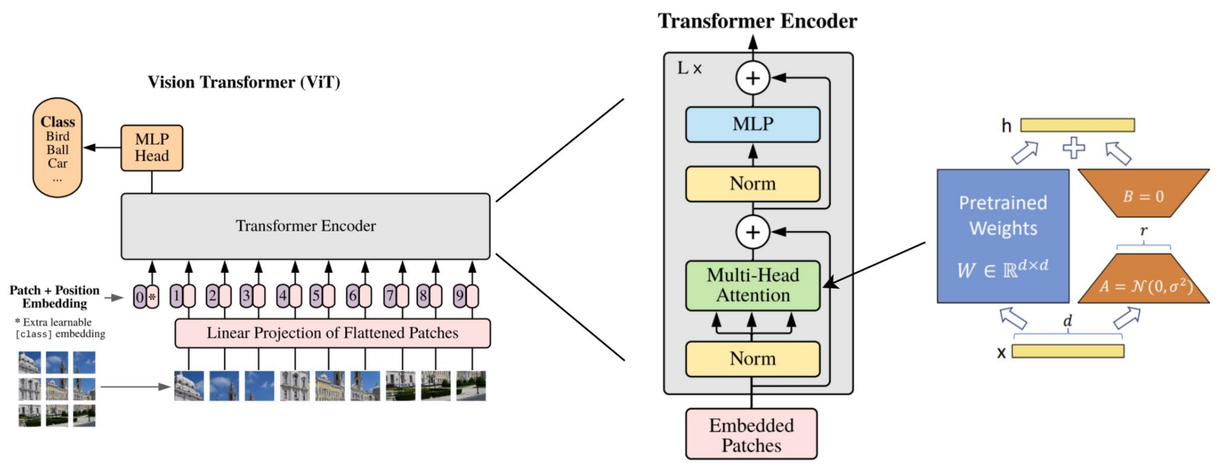
[1] Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., & Joulin, A. (2021). Emerging Properties in Self-Supervised Vision Transformers. ArXiv. https://arxiv.org/abs/2104.14294
[2] Hu, E. J., Shen, Y., Wallis, P., Li, Y., Wang, S., Wang, L., & Chen, W. (2021). LoRA: Low-Rank Adaptation of Large Language Models. ArXiv. https://arxiv.org/abs/2106.09685
[3] Kirkpatrick, J., Pascanu, R., Rabinowitz, N., Veness, J., Desjardins, G., Rusu, A. A., Milan, K., Quan, J., Ramalho, T., Hassabis, D., Clopath, C., Kumaran, D., & Hadsell, R. (2016). Overcoming catastrophic forgetting in neural networks. ArXiv. https://doi.org/10.1073/pnas.1611835114
[4] Liu, Y., Kothari, P., van Delft, B., Bellot-Gurlet, B., Mordan, T., & Alahi, A. (2021). TTT++: When does self-supervised test-time training fail or thrive? *Advances in Neural Information Processing Systems, 34*, 21808–21820.

Methodology

We perform test-time fine-tuning in 3 ways.
- Naive Fine-tuning (FT)
- Elastic Weight Consolidation (EWC)
- Low-Rank Adaptation (LoRA)







Experimental Results

Observing distribution shift when fitting nearest-neighbor KNN

CIFAR10 classification results.
Val and Test show Top1 accuracy for 10NN and 20NN classifiers.

Method	Val		Test	
	10NN	20NN	10NN	20NN
Base	87.16	87.19	76.70	76.75

Camelyon17 classification results.
Harmful and Not-Harmful show Top1 accuracy for 10NN and 20NN classifiers.

Method	Harmful		Not-Harmful	
	10NN	20NN	10NN	20NN
Base	73.19	74.22	78.90	79.28

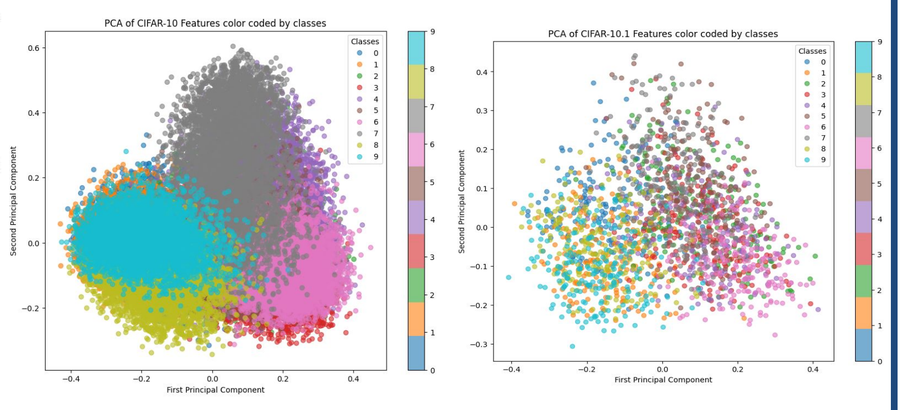
Naive fine-tuning results

CIFAR10 classification results with Naive finetuning.

Method	Val	Test
Base	87.16	75.2
First iteration of finetuning	87.12	75.0
Second iteration of finetuning	86.54	74.8
Third iteration of finetuning	85.96	74.63
Fourth iteration of finetuning	85.40	74.13

Naive FT Camelyon17 classification results.

# Epochs	Harmful		Not-Harmful	
	10NN	20NN	10NN	20NN
10 Epochs	73.21	74.24	79.03	79.36
20 Epochs	73.23	74.26	79.18	79.46
30 Epochs	**73.25**	74.27	79.25	79.53
40 Epochs	73.24	74.28	**79.27**	79.56
50 Epochs	73.24	**74.29**	**79.27**	**79.57**



LoRA fine-tuning Results

CIFAR10 classification results with LoRA.

Method	Val	Test
Base	87.1	76.7
Rank-4 LoRA	86.5	75.6
Rank-8 LoRA	85.8	75.0

EWC Results

EWC Camelyon17 classification results.

# Epochs	Harmful		Not-Harmful	
	10NN	20NN	10NN	20NN
10 Epochs	73.21	74.26	79.02	79.36
20 Epochs	73.27	74.28	79.19	79.44
30 Epochs	73.30	74.32	79.23	79.51
40 Epochs	73.24	74.28	**79.27**	**79.56**
50 Epochs	**73.31**	**74.33**	79.26	**79.56**

Conclusion:

We find that naive fine-tuning, LoRA, and EWC either negatively impact the model (in the case of a high # of classes task like CIFAR-10/10.1) or only slightly improve model performance in the binary classification case of CAMELYON. This implies that SSL-pretrained ViTs can't be simply fine-tuned by retraining the way they were trained, and instead require a more complex methodology.

It's also known that ViTs also traditionally require a lot of data, and our fine-tuning datasets are disproportionately small to our training datasets. This discrepancy could be another source of why we cannot fine-tune ViTs this way