Neural Network Linearization for Out-of-Distribution Robustness

James Campbell[1], Yaodong Yu[2], Yi Ma[2]

1 Cornell University; 2 University of California, Berkeley

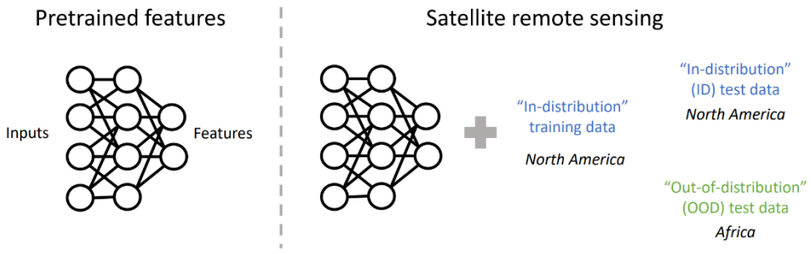
Background

Foundation Models - In recent years, deep learning has shifted towards a paradigm where practitioners will download large models that have been pretrained on massive, diverse data. Once downloaded, these models will usually be "fine-tuned" on a specific dataset and task.

Distribution Shifts – Machine learning models perform well in their training distribution, but often fail catastrophically when exposed to inputs that are slightly different in nature. In order to promote the safe adoption of AI, it is important that ML systems work in a variety of realistic scenarios that they may encounter in the real world.

Fine-Tuning vs Linear Probing – The two most common ways of adapting a pretrained model to a downstream task is fine-tuning, where all of the parameters are re-trained on the new data, and linear probing, where only the parameters in the last layer are updated. Prior work has shown that while fine-tuning works best in-distribution, it is less robust than linear probing and can distort pretrained features.

Problem Setting



Approach: The eNTK

The Neural Tangent Kernel (NTK) – The neural tangent kernel is found by taking the first-order Taylor expansion of a neural network with respect to its parameters. Concretely, the NTK is simply the Jacobian of the network—a matrix containing the partial derivatives of the network's outputs with respect to each of its parameters. In the theory of deep learning, it is possible to prove many theoretical results about neural networks using the NTK, though these often rely on several unrealistic assumptions.

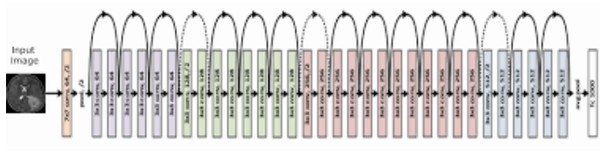
$$f(\theta, x) \approx f(\theta_0, x) + \nabla_\theta f(\theta_0, x)^\top (\theta - \theta_0)$$

$$\phi(x) = \nabla_\theta f(\theta_0, x)$$

$$f(\theta, x) \approx \theta^\top \phi(x)$$

The Empirical NTK (eNTK) – Putting its theoretical use aside, it is possible to linearize neural networks in practice by computing their NTK's, which is in this setting called the empirical neural tangent kernel (eNTK). In order to do this efficiently, we make several approximations. First, because we start with a pretrained model whose last layer will be randomly initialized, we reduce the network to only one output dimension. This reduces the Jacobian to simply the gradient with respect to this single output. Intuitively, this approximation holds because the randomly initialized output layer doesn't contain any information to distinguish one output from any other.

The second approximation we make is to subsample 500,000 parameters of the network and to only compute derivatives with respect to those. Preliminary experiments suggest that this approximation is valid as there is little gains to be had by improving the parameter dimension beyond 500,000. For all of our experiments conducted so far, we have used a pre-trained ResNet-50 as our network, meaning our sample of 500,000 represents about 2% of the networks ~23,000,000 total parameters. In the future, we hope to conduct further testing on how this scales to larger pre-trained models, including those that are popular in NLP.



Once we obtain the 500,000-dimensional NTK representations for a given dataset, we compute the NTK *matrix* given by the data matrix's inner product. This gives us a nxn kernel matrix, where n is the size of the dataset. From there, we employ the kernel trick to solve for the parameters in linear regression. We solve this convex problem using the following update rule (derived below), inserting it into Nestorov's Accelerate Gradient to speed up convergence.

$$L = ||X\theta - Y||^2 + \beta||\theta||^2$$

$$\Delta\theta = -\nabla_\theta L \propto -X^T(X\theta - Y) + \beta\theta$$

$$\theta = X^\top\alpha;\ K = XX^\top$$

$$\Delta\theta = X^\top\Delta\alpha = -\nabla_\theta L \propto -X^T\left((K\alpha - Y) + \beta\alpha\right)$$

$$\Delta\alpha \propto -(K\alpha - Y) - \beta\alpha$$

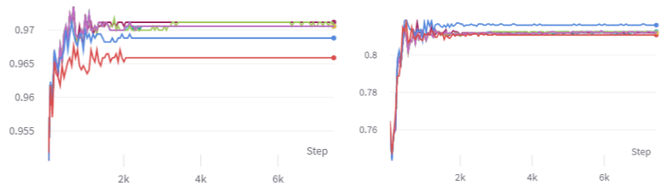
We report results for both early stopping on the in-distribution validation set and for running gradient descent until convergence. We conduct a telescopic search over L2 regularization strengths, finding that results are relatively robust to the level of regularization.

Experiments

So far, we obtained results for Living-17, which is a distribution shift benchmark sampled from ImageNet. We compare our eNTK results to those reported in [1] and find that the eNTK is better both in- and out-of-distribution compared to traditional fine-tuning.

Living-17	ID	OOD
Scratch	92.4%	58.2%
Linear Probing	96.5%	**82.2%**
Fine-Tuning	97.1%	77.7%
eNTK (early stopping)	**97.4%**	81.2%
eNTK (converged)	97.1%	81.3%

In- and out-of-distribution accuracy during training and across different regularization strengths



Ongoing and Future Work

We're currently working on getting results for other datasets and models, both in vision and NLP. Moreover, several ablation studies ought to be performed, investigating different approximations, sampling strategies, and theoretical hypotheses, such as how the eNTK performs specifically on data outside the span of the training data.

References

[1] Ananya Kumar, Aditi Raghunathan, Robbie Jones, Tengyu Ma, & Percy Liang. Fine-Tuning can Distort Pretrained Features and Underperform Out-of-Distribution. ICLR 2022.

[2] Alexander Wei, Wei Hu, & Jacob Steinhardt. More Than a Toy: Random Matrix Models Predict How Real-World Neural Representations Generalize. ICML 2022.

Acknowledgements

We would like to thank the NSF and SUPERB REU program for enabling this project, especially Leslie Mach. James would also like to thank the countless people he's met at UC Berkeley for a wonderful and intellectually-enriching summer!