Research Statement

Suraj Srinivas
Harvard University
ssrinivas@seas.harvard.edu

Modern progress in machine learning is primarily driven by large neural network architectures that are capable of ingesting large amounts of data. While this paradiagm has led to impressive advances in predictive performance across a range of tasks and domains, the sheer size and complexity of models hampers our ability to understand, improve and iterative over them. To this end, my research focusses on simplifying neural network models along two different axes - computational and functional. While **computational** simplicity of neural networks focusses on having a small structural footprint in terms of the number of weights or neurons, **functional** simplicity involves exhibiting predictable behaviour that can help us ultimately verify whether models align with human preferences.

Prior Work

Computationally Efficient Deep Learning

Standard neural networks are highly redundant, making it possible to prune a large number of weight or neurons without sacrificing predictive accuracy. We showed an early proof-of-concept of this phenomenon by identifying and pruning only duplicate neurons in standard pre-trained vision models (Srinivas and Babu, 2015). This led to the question: **what are optimal methods to perform model pruning** that are able to identify all sources of parameter redundancy? In general this is intractable as the underlying pruning problem is discrete in nature, and is incompatible with gradient-based optimizers used in practice obtain optimal real-valued weight parameters. In my research, I explored different strategies to solve this challenging problem.

Stochastic Continuous Relaxation We first proposed a pruning algorithm that relaxes the intractable discrete optimization problem into a stochastic continuous one. Specifically, we automatically prune weights / neurons by introducing multiplicative binary stochastic gate variables with each unit and corresponding model complexity regularizers that encourage the total number of such units to be small (Srinivas and Babu, 2016b; Srinivas et al., 2017). Finally, we connected this pruning approach with a popular model regularization technique called dropout by defining a general class of regularizers called *generalized dropout* (Srinivas and Babu, 2016a) which contains both dropout and model pruning as special cases.

ℓ_0 Regularization Another solution to the pruning problem is to view this through the lens of parameter sparsity and ℓ_0 regularization. A classic solution to the problem of inducing sparsity is via iterative hard thresholding, or projected gradient descent (PGD). We first observed that common pruning heuristics such as iterative magnitude pruning, implement an approximate version of PGD. We propose to improve upon this by using **cyclical pruning** (Srinivas et al., 2022a), an algorithm that better approximates the optimal PGD solution and obtains improved compression ratios as a result.

Submodular Optimization As an alternative to relaxing the discrete problem, another approach is to embrace this discrete nature and use discrete optimization algorithms directly. In our work (Halabi et al., 2022), we observe that the layerwise pruning problem admits a weakly submodular structure, and that one can use efficient greedy subset selection algorithms to provably approximate the oracle pruning solution.

Local Geometric Analysis of Deep Models

Robustness and local post-hoc interpretability of neural networks both depend on the local geometry of the input-output map. For example, the more locally constant the model, the more robust it is, and the more locally linear the model, the more 'useful' are its gradient maps. However, these local geometric properties are under-utilized while training neural network models, leaving models non-robust and uninterpretable. We thus propose to analyze and regularize such local model geometry of practical deep models to improve their robustness and interpretability properties.

Post-hoc Interpretability We show that for post-hoc interpretability, particularly feature attribution methods, several popular methods essentially perform local function approximation with a linear model (Han et al., 2022). This indicates that the problem of feature attribution is intricately linked to that of capturing local model geometry.

However, while canonical methods such as input-gradients capture local information, they fail to produce attributions at flat input regions. We show that is a fundamental trade-off which occurs due to their limited expressivity. To alleviate this, we proposed the full-gradient representation (Srinivas and Fleuret, 2019), which is more expressive than canonical feature attribution methods. Using this, we also proposed an approximate saliency map called FullGrad which incorporates gradient information from all layers of a deep model, which we found to outperform other saliency maps from literature, especially for large off-the-shelf vision models.

An important yet unexplored question in this area is the question of **what causes structure in local geometry of deep models** in a manner that is useful for interpretability, especially when such structure is unnecessary for generalization (Srinivas and Fleuret, 2021). We provided an explanation by invoking the latent energy-based generative properties $p(x \mid y)$ of softmax-based discriminate models $p(y \mid x)$, and showed that the training of this latent generative model specifically using score-matching (Hyvärinen, 2005), dictates gradient interpretability. Overall, this work shows that commonly used input-gradient methods do not capture information related to the discriminative model $p(y \mid x)$, but a latent generative model $p(x \mid y)$, and we must rethink its usage in practice.

Robust Distillation While standard robust models often sacrifice predictive accuracy, this is not the case for the task of distillation, where we showed that ideas from robustness can make distillation more sample efficient (Srinivas and Fleuret, 2018). Specifically we showed that Jacobian information at data points helps capture the local neighborhood around that data point, and thus provides more information to the student model to learn from the teacher model. We also showed applications in transfer learning using the same technique, which had similar benefits.

Low-Curvature Models for Robustness Our work on explaining the structure of local geometry of deep models (Srinivas and Fleuret, 2021) also showed that robust models have similar generative modelling properties to model trained explicitly with generative regularizers. To explore this further, we investigated the inductive biases of robust models, and found that robust models tend often to have smaller curvature (in terms of Hessian norms) than non-robust ones (Srinivas et al., 2022b), and We used this observation to propose new ways to training robust models based explicitly penalizing an efficient upper bound on model curvature.

Future Work

In the future, I would like to focus my research toward novel model architectures that improve data and compute efficiency of models. In addition, large models require improved interpretability methods, as current paradiagms have been shown to be conceptually ineffective (Lipton, 2018) to address questions involving spurious correlations and precisely characterizing model behaviour. Toward this end, I propose three broad directions for future work: active inference, joint density modelling and structure-driven interpretability.

Adaptive Inference Neural Networks

A significant shortcoming of standard neural architectures is that their inference is **passive**, in other words, the computation for a given input is fixed. This makes them computationally inefficient, as they allocate equal compute to all inputs irrespective of their relative difficulty. This restricts our ability to scale model training given finite computational resources. To overcome this fundamental limitation, **I intend to develop compute efficient models with active inference**. There are several ways to implement such adaptive mechanisms. One way is to view layer-wise computation in neural networks as iterates of a dynamical system tending to converge toward some fixed point, similar to deep equilibrium models (Bai et al., 2019) and neural ODEs (Chen et al., 2018). Alternately, one may learn the stopping criteria via gradient descent similar to Graves (2016). Another direction is to implement 'saccades', i.e., a sequence of local attention masks on the input such that the prediction may be performed with the smallest number of such saccades, similar to Mnih et al. (2014). There are several potential advantages to such classes of models:

- **Simulate Large Deep Models**: Similar to the way deep models with a small number of parameters are able to simulate exponentially wide neural networks (Poole et al., 2016), small adaptive neural networks are also able to simulate much larger deep neural networks. The principle here is that the underlying function can be encoded via the fixed points or trajectories of a dynamical system, and that simple parameterization can encode an arbitrarily complex dynamics. (E.g.: the logistic map, and other chaotic systems)
- **Practical 1-Lipschitz Models**: A well-known flaw with standard deep models is their large Lipschitz constants which leads to a lack of robustness. One hypothesis for this is that fitting robust models requires large number of parameters (of the order of input dimensionality) (Bubeck and Sellke, 2021). Adaptive inference models may provide a solution to this by simulating large models with a small-Lipschitz constant, thus offering robustness and good predictive performance simultaneously.
- **Algorithmic Reasoning**: Many algorithmic reasoning tasks can be solved via iterative algorithms, for which dynamical models are better suited than static neural network models.

While previous works provide proofs of concept that adaptive computation is possible in practice, it remains to be shown that they can lead to practical efficiency gains on real benchmarks precisely by simulating large deep models.

Joint Density Models for Efficient Data Use

A goal highly related to the above is to build predictive models that learn the joint density $p(y, x)$ rather than simply the discriminative component $p(y \mid x)$, which makes better use of available data. This has strong parallels with our earlier work on implicit generative modelling in discriminative models (Srinivas and Fleuret, 2021) using the principle of score-matching (Hyvärinen, 2005). Learning the joint density implies advantages for a multitude of tasks such as **robustness**, **OOD detection**, **uncertainty estimation** and interpretability (Grathwohl et al., 2020). In other words, learning $p(x \mid y)$ in addition to $p(y \mid x)$ serves as a regularizer enabling it to generalize better on small training sets. While current methods to train such joint density models are based on contrastive divergence (Grathwohl et al., 2020) or score-matching (Srinivas and Fleuret, 2021), it is an open question whether there exists such scalable and efficient generative modelling principles which can be generally applicable.

Interpretability via Function Approximation

Our recent work (Han et al., 2022) shows that several post hoc interpretation methods can all be thought of via one unifying principle of local function approximation. This principle can serve as a general principle of interpretability beyond simply local post hoc explanations. While in initial work (Han et al., 2022) we considered the case of explaining individual data points using a linear model, we can consider explaining several data points at once, and indeed, approximate them using other modelling choices. In addition, one may use this framework not just to explain an individual model, but also learning algorithms locally. Indeed, this design space is largely unexplored and can form the basis for future work. This line of work also informs model designers regarding modelling choices - one must not choose an arbitrary black-box model class, but one that can be readily explained via function approximation.

Overall, these new lines of research can lead to a new generation of neural network models that are simultaneously more compute efficient and interpretable, enabling us to further scale large models while still being able to provide insight about their behaviour.

References

Bai, S., Kolter, J. Z., and Koltun, V. (2019). Deep equilibrium models. *Advances in Neural Information Processing Systems*, 32.

Bubeck, S. and Sellke, M. (2021). A universal law of robustness via isoperimetry. In Beygelzimer, A., Dauphin, Y., Liang, P., and Vaughan, J. W., editors, *Advances in Neural Information Processing Systems*.

Chen, R. T., Rubanova, Y., Bettencourt, J., and Duvenaud, D. K. (2018). Neural ordinary differential equations. *Advances in neural information processing systems*, 31.

Grathwohl, W., Wang, K.-C., Jacobsen, J.-H., Duvenaud, D., Norouzi, M., and Swersky, K. (2020). Your classifier is secretly an energy based model and you should treat it like one. In *International Conference on Learning Representations*.

Graves, A. (2016). Adaptive computation time for recurrent neural networks. *arXiv preprint arXiv:1603.08983*.

Halabi, M. E., Srinivas, S., and Lacoste-Julien, S. (2022). Data-efficient structured pruning via submodular optimization. In Oh, A. H., Agarwal, A., Belgrave, D., and Cho, K., editors, *Advances in Neural Information Processing Systems*.

Han, T., Srinivas, S., and Lakkaraju, H. (2022). Which explanation should i choose? a function approximation perspective to characterizing post hoc explanations. In Oh, A. H., Agarwal, A., Belgrave, D., and Cho, K., editors, *Advances in Neural Information Processing Systems*.

Hyvärinen, A. (2005). Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6(Apr):695–709.

Lipton, Z. C. (2018). The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery. *Queue*, 16(3):31–57.

Mnih, V., Heess, N., Graves, A., et al. (2014). Recurrent models of visual attention. *Advances in neural information processing systems*, 27.

Poole, B., Lahiri, S., Raghu, M., Sohl-Dickstein, J., and Ganguli, S. (2016). Exponential expressivity in deep neural networks through transient chaos. In Lee, D., Sugiyama, M., Luxburg, U., Guyon, I., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc.

Srinivas, S. and Babu, R. V. (2015). Data-free parameter pruning for deep neural networks. In *Proceedings of the British Machine Vision Conference 2015 (BMVC 2015)*, pages 7–10.

Srinivas, S. and Babu, R. V. (2016a). Generalized dropout. *arXiv preprint arXiv:1611.06791*.

Srinivas, S. and Babu, R. V. (2016b). Learning the architecture of deep neural networks. In *Proceedings of the British Machine Vision Conference 2016 (BMVC 2016)*.

Srinivas, S. and Fleuret, F. (2018). Knowledge transfer with Jacobian matching. In Dy, J. and Krause, A., editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 4723–4731. PMLR.

Srinivas, S. and Fleuret, F. (2019). Full-gradient representation for neural network visualization. In *Advances in Neural Information Processing Systems*, pages 4126–4135.

Srinivas, S. and Fleuret, F. (2021). Rethinking the role of gradient-based attribution methods for model interpretability. In *International Conference on Learning Representations*.

Srinivas, S., Kuzmin, A., Nagel, M., van Baalen, M., Skliar, A., and Blankevoort, T. (2022a). Cyclical pruning for sparse neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 2762–2771.

Srinivas, S., Matoba, K., Lakkaraju, H., and Fleuret, F. (2022b). Efficient training of low-curvature neural networks. In Oh, A. H., Agarwal, A., Belgrave, D., and Cho, K., editors, *Advances in Neural Information Processing Systems*.

Srinivas, S., Subramanya, A., and Venkatesh Babu, R. (2017). Training sparse neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*.