Fundamental Mathematical Tools on the Path to AGI

Amazing advances over the last many years have brought us to a place where the community is moving its focus from artificial intelligence or AI to artificial *general* intelligence or AGI. While many definitions exist, the following is the most interesting distinction to me between AI and AGI – In AI, we were focused on building systems that could perform a single skill (such as classifying images, predicting sentiment, playing specific games) given enough data and computational resources; the focus of AGI is (or should be in my opinion) to build systems that can *acquire new skills* efficiently both in terms of computational and data resources. This is especially relevant in scientific domains where data collection can be extremely expensive, both in terms of monetary costs and manual effort. The paradigm shift from AI to AGI requires the core problem of "learning a skill" to evolve into the problem of "learning to acquire new skills efficiently".

The "Bitter Lesson" in AI[1] is that, up until now, simplistic models, equipped with lots of data and brute-force computation, will often beat more sophisticated models. Motivated by this Bitter Lesson, there is significant effort in building "large" models utilizing the computational power of big compute clusters and all the data available on the internet, and this has demonstrated varying degrees of successes in AGI. Unfortunately, these massive requirements are leading us to societal problems where computation and data at this scale is only available to a few, leading to a form of tech-oligarchy, and we are progressively consuming more fossil fuels at a time when we need to be more strategic in our power consumption. Furthermore, at a time when the compute hardware is unable to keep up with the computation needs of the AI community in this post Moore's Law world[2], it seems counterproductive to utilize compute blindly to improve performance – this might be the final endgame of brute-force computation, and we need to reassess the bitter lesson. *The primary pursuit of my research is to demonstrate similar or better AGI capabilities much more efficiently (both in terms of computation and data) without the current practice of prohibitive (and possibly wasteful) use of resources*. With this goal, I focus my research on the following three high-level questions in this new paradigm:

- What are the right optimization objectives for this learning problem, and how can we solve these optimization problems efficiently?
- What are the right types of models for this paradigm and how big do these models really need to be?
- How can we leverage the inherent structure in the data to reduce the training data requirements?

I will briefly elaborate on each of these questions, and my research along these lines in the following.

What is the right learning objective?

Given a set of training examples, the standard practice in AI for "learning a skill" involves selecting a class of models and a loss function, and starting from some initial model in that class, performing the following steps iteratively until some stopping criteria (such as computational budget or desired accuracy) is triggered: (i) Predict on (a subset of) the training data, (ii) Compute the loss on these predictions, and (iii) Update the current model to reduce this loss. This general procedure (depicted in Fig 1) is applicable to differentiable models such as neural networks, where the model update utilizes some form of gradient descent, as well as to more discrete models such as decision trees and program synthesis, where the model update utilizes combinatorial search. Regularization is often used to subvert overfitting to the training data, but can be achieved explicitly with the choice of the model class or loss functions, or implicitly through specific model update procedures. This "predict, compute loss, update model" cycle finally produces a model for the particular skill at hand, and the goal is to ensure that this model generalizes on unseen examples. Given enough training examples, expressive enough models, and sufficient computational resources, this procedure has been quite successful.

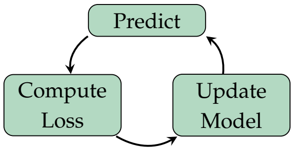

Figure 1: Learning a skill

However, it is not clear how this procedure is directly applicable to the problem of "learning to acquire new skills". Standard approach has been to train a large (often autoregressive) model with a large and varied training data (corresponding to a variety of skills) and some auxilliary loss functions (such as next-token prediction). These models are then fine-tuned for a variety of tasks (such as safety alignment) via techniques such as reinforcement learning or preference optimization on specialized data (such as human feedback, chains of thought). For any new skill, the skill acquisition step can involve (i) fine-tuning (with adaptors like LoRA) if enough skill-specific data is available, (ii) in-context learning where skill-specific data is part of the input to the model, and (iii) test-time scaling where the model generates multiple outputs (potentially recursively).

Yet, this staged approach with (often hand-crafted) stage-specific data and loss function does not directly fit in the tried-and-tested "predict, compute loss, update model" cycle for the problem of "learning to acquire new skills". Given a set of skills (and available skill-specific data), an alternate learning process could extend the above cycle to "acquire skill & predict, compute loss, update model" (see Fig 2), where the "predict" step (in traditional learning) is replaced with a step where the model first acquires the skill, and then makes subsequent skill specific predictions to compute the loss. As the skill acquisition itself is often a mini-learning problem, it makes the overall learning process a *bilevel optimization problem* where the outer level problem is to learn to acquire new skills while the inner level problem is the acquistion of specific skills. Bilevel optimization is closely related to Stackelberg games, and their solutions often correspond to the a perfect Nash Equilibria of a specific game. While general bilevel optimization is an extremely challenging problem, I have already shown various applications of it in AI and machine learning such as meta-learning[3;4], representation learning[5], hyperparameter optimization[6–10], model pruning[11], clustering[12], unlearning[13;14] and many more[15;16]. In addition to appropriately addressing and leveraging the bilevel nature of the problem, *skill generalization* – ability to generalize to unseen new skills – requires our explicit attention. In our paper **Min-max Multi-objective Bilevel Optimization with Applications in Robust Machine Learning**, we provide preliminary theoretical and empirical evidence which demonstrate that *robust multi-objective bilevel optimization*[17;18] is one way to promote generalization to unseen skills.

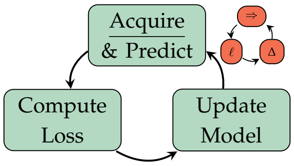

Figure 2: Learning to acquire new skills

I believe that bilevel optimization provides a powerful framework for the path to AGI as it natively allows expression of a lot of "meta" processes necessary for AGI, and one goal of my research is to scale bilevel optimization (and its robust multi-objective variants) to levels comparable to standard *single-level* optimization via the ubiquitous yet simple stochastic gradient descent and its variants.

What are the right models?

The current trend in AI is to build larger and larger models (or "scale up") with the promise of better expressivity. Models are made deeper to endow them with more levels of information processing, which is useful for acquiring and predicting for harder skills. However, the processing depth of these models are fixed, regardless of the hardness of the problem at hand (which inevitably results in a push to build deeper and deeper models as we want these models to be ready to solve hard problems). I believe this trend of larger models is not sustainable and also not necessary (as evidenced with various distilled versions of these models that are orders of magnitude smaller and yet equally capable). In contrast, there are models in literature such as those depending on a parse tree (implicitly or explicitly[19]) for natural language processing, and energy based models (utilizing some form of fixed-point iterations), that adapt the prediction/inference processing time to the input without increasing the size of the model – larger/harder problems requiring more recursive processing (compare the 2-digit multiplication in Fig 3 to the harder 3-digit multiplication in Fig 4).

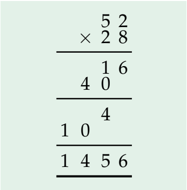

Figure 3: 2 digit multiplication

Associative Memory networks, one of the very first "neural networks" (which were recently recognized with the 2024 Nobel Prize in Physics), and their modern variant Dense Associative Memories[20] are one such class of energy-based models, where the models are parameterized with "memory vectors" and the prediction process performs a gradient descent over an energy function, with harder inputs requiring more descent steps, thereby being adaptive to the input. The expressivity of these models are tied to their "memory capacity", with larger capacity implying more expressivity but also larger models (as the model size scales linearly with the number of memories), and scaling up to exponential (in the input dimension) memory capacity guarantees universal approximation with an exponentially large model. Utilizing techniques from Fourier analysis and kernel methods, and leveraging the underlying problem structure, we disentangle the model size from the number of memories[21] in our paper **Dense As-**

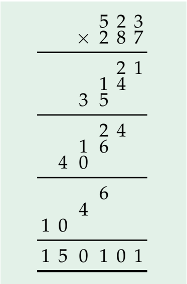

Figure 4: 3 digit multiplication

sociative Memories Through the Lens of Random Features, demonstrating that there are ways to have the capabilities of a large model without having to scale up the actual model.

> I believe that such models will be very useful on the path to AGI, and another goal of my research is to utilize core mathematical tools to develop such models with extremely favorable size-expressivity trade-offs, thereby significantly reducing the computational overhead of training and predicting with these models.

How to reduce training data requirements?

As the large models are being trained on troves of data – openly available, proprietary, human generated, model generated and human evaluated – the computational costs have become prohibitive as have the data requirements. Furthermore, to improve the performance of these models on reasoning tasks, more and more prompt augmentation data (such as derivation traces) are generated for training. However, most problems (including reasoning ones) often have a hierarchical *compositional* structure (albeit often not entirely explicit) – "*the meaning of the whole is a function of the meanings of the parts and of the way they are syntactically combined*". Examples includes problems with a grammar based language, mathematical operations such as long addition or long multiplication (see Figs 3 & 4 where the final solution is hierarchically built up utilizing single digit multiplication, multiplication by 10, and long addition), and any reasoning tasks based on (implicit or explicit) algebraic expressions[22]. In such problems, the answer is often built up recursively with longer/harder problems requiring deeper recursions (highlighting the aforementioned need for models that can adapt to the hardness of the problem at hand). The compositional structure allows us to solve different or harder problems by breaking the problem into smaller easier known pieces, and solving these easier problems based on our learning. Compositional generalization is a core tenet of cognitive science developed by linguists, and a key to the ability of humans to learn from extremely small number of examples). Of course, it is not necessary that AGI models solve problems in the same way as we humans tackle them. However, this inherent compositional structure, if properly leveraged, allows us break each example problem into multiple (easier to learn) problems, thereby both reducing the hardness of what needs to be learned, and providing more examples to learn these easier problem from. This also requires us to learn how to break problems up and put solutions back together hierarchically, which is nontrivial.

A first step towards leveraging this compositional structure is formally characterizing this notion, something absent in the literature. We present on novel and precise characterization in our paper[23] **What makes Models Compositional? A Theoretical View**. We explicitly tease out the different factors affecting the *compositional complexity* of general functions, and of standard AI models such as recurrent and convolutional neural networks, and transformers. The results highlight that most existing models have extremely high compositional complexity disincentivizing these models from learning simple compositional functions, yet, these same models lack the adaptivity that is necessary to learn the underlying compositional structure – these models end up learning complex functions to fit the training data as they lack the expressivity to break the problems up into (problem-dependent) simpler pieces – a somewhat counterintuitive result. Based on this theoretical study, we develop simple inductive biases that allow these off-the-shelf models (such as transformers) to learn reasoning tasks significantly faster with far few examples than the standard counterparts while achieving the same level of generalization[24].

> I believe that this inherent, though implicit, compositional structure of many problems can significantly reduce the data requirements on the path to AGI, and a goal of my research is to develop models and training procedures that are able to leverage this structure, thereby reducing both the amount of data needed as well as the associated computational costs of processing such data.

Further Research Interests

I have a wide range of research interests, and I am always excited to learn about new areas and make connections between different areas. In this process, I have spent time focusing on the following areas: Optimization (single and bilevel), Automated Machine Learning & Data Science, Large Scale Learning, Computational Geometry, Efficient All-Pairs Algorithms & Analysis, Density Estimation, Kernel Methods, Associative Memories & Energy-based Models, Machine Unlearning, Sparse Learning, Neuro-inspired Learning, Compositional Generalization. Please see my resume here.

References

[1] Richard Sutton. The bitter lesson. *Incomplete Ideas (blog)*, 13(1):38, 2019.

[2] Audrey Woods. The death of moore's law: What it means and what might fill the gap going forward. *MIT CSAIL*, 2024.

[3] Chen Fan, Parikshit Ram, and Sijia Liu. Sign-MAML: Efficient model-agnostic meta-learning by SignSGD. In *5th NeurIPS 2021 Workshop on Meta-Learning*, 2021.

[4] Alex Gu, Songtao Lu, Parikshit Ram, and Lily Weng. Nonconvex min-max bilevel optimization for task robust meta learning. In *ICML 2021 Workshop on Beyond first-order methods in ML systems*, 2021.

[5] Yihua Zhang, Pranay Sharma, Parikshit Ram, Mingyi Hong, Kush Varshney, and Sijia Liu. What is missing in irm training and evaluation? challenges and solutions. In *The 11th International Conference on Learning Representations*, 2023.

[6] Sijia Liu, Parikshit Ram, Deepak Vijaykeerthy, Djallel Bouneffouf, Gregory Bramble, Horst Samulowitz, Dakuo Wang, Andrew Conn, and Alexander Gray. An admm based framework for automl pipeline configuration. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 4892–4899, 2020.

[7] Parikshit Ram. On the optimality gap of warm-started hyperparameter optimization. In *International Conference on Automated Machine Learning*, pages 12–1. PMLR, 2022.

[8] Yi Zhou, Parikshit Ram, Theodoros Salonidis, Nathalie Baracaldo, Horst Samulowitz, and Heiko Ludwig. Single-shot general hyper-parameter optimization for federated learning. In *The 11th International Conference on Learning Representations*, 2023.

[9] Parikshit Ram, Alexander G Gray, Horst C Samulowitz, and Gregory Bramble. Toward theoretical guidance for two common questions in practical cross-validation based hyperparameter selection. In *Proceedings of the 2023 SIAM International Conference on Data Mining (SDM)*, pages 802–810. Society for Industrial and Applied Mathematics, 2023.

[10] Yi Zhou, Parikshit Ram, Theodoros Salonidis, Nathalie Baracaldo, Horst Samulowitz, and Heiko Ludwig. Hyper-parameter optimization in federated learning. In *Federated Learning*, pages 237–255. Academic Press, 2024.

[11] Yihua Zhang, Yuguang Yao, Parikshit Ram, Pu Zhao, Tianlong Chen, Mingyi Hong, Yanzhi Wang, and Sijia Liu. Advancing model pruning via bi-level optimization. *Advances in Neural Information Processing Systems*, 35:18309–18326, 2022.

[12] Bishwajit Saha, Dmitry Krotov, Mohammed J Zaki, and Parikshit Ram. End-to-end differentiable clustering with associative memories. In *International Conference on Machine Learning*, pages 29649–29670. PMLR, 2023.

[13] Jiancheng Liu, Parikshit Ram, Yuguang Yao, Gaowen Liu, Yang Liu, Pranay Sharma, Sijia Liu, et al. Model sparsity can simplify machine unlearning. *Advances in Neural Information Processing Systems*, 36, 2023.

[14] Jinghan Jia, Jiancheng Liu, Yihua Zhang, Parikshit Ram, Nathalie Baracaldo, and Sijia Liu. Wagle: Strategic weight attribution for effective and modular unlearning in large language models. *Advances in Neural Information Processing Systems*, 37, 2024.

[15] Momin Abbas, Yi Zhou, Nathalie Baracaldo, Horst Samulowitz, Parikshit Ram, and Theodoros Salonidis. Byzantine-resilient bilevel federated learning. In *2024 IEEE 13rd Sensor Array and Multichannel Signal Processing Workshop (SAM)*, pages 1–5. IEEE, 2024.

[16] Akihiro Kishimoto, Djallel Bouneffouf, Radu Marinescu, Parikshit Ram, Ambrish Rawat, Martin Wistuba, Paulito Palmes, and Adi Botea. Bandit limited discrepancy search and application to machine learning pipeline optimization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 10228–10237, 2022.

[17] Alex Gu, Songtao Lu, Parikshit Ram, and Lily Weng. Robust multi-objective bilevel optimization with applications in machine learning. In *INFORMS Annual Meeting*, 2022.

[18] Alex Gu, Songtao Lu, Parikshit Ram, and Tsui-Wei Weng. Min-max multi-objective bilevel optimization with applications in robust machine learning. In *The 11th International Conference on Learning Representations*, 2023.

[19] Tim Klinger, Luke Liu, Soham Dan, Maxwell Crouse, Parikshit Ram, and Alexander Gray. Compositional program generation for systematic generalization. In *IJCAI 2023 Workshop on Knowledge-Based Compositional Generalization*, 2023.

[20] Dmitry Krotov and John J Hopfield. Dense associative memory for pattern recognition. *Advances in neural information processing systems*, 29, 2016.

[21] Benjamin Hoover, Duen Horng Chau, Hendrik Strobelt, Parikshit Ram, and Dmitry Krotov. Dense associative memory through the lens of random features. *Advances in Neural Information Processing Systems*, 37, 2024.

[22] Takuya Ito, Murray Campbell, Lior Horesh, Tim Klinger, and Parikshit Ram. Quantifying artificial intelligence through algebraic generalization. *arXiv preprint arXiv:2411.05943*, 2024.

[23] Parikshit Ram, Tim Klinger, and Alexander G Gray. What makes models compositional? a theoretical view. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence (IJCAI-24)*, 2024.

[24] Parikshit Ram, Kenneth L Clarkson, Shashanka Ubaru, Tim Klinger, and Alexander G Gray. A theoretical view of sparse attention transformers. *arXiv*, 2025.