Caltech
Student-Faculty Programs

Session K Abstracts

Synthetic data-driven knowledge integration for large language models
Amrita Pasupathy
Mentors: Yisong Yue and Lauren Hyoseo Yoon

Although large language models (LLMs) are powerful tools for accessing information, they often struggle to incorporate new knowledge acquired after pretraining without expensive retraining. To address this issue, we propose a knowledge integration framework that can add new information to a base model while also preserving the ability to incorporate future information incrementally. Specifically, we construct a synthetic data engine that transforms the original data into synthetic variants designed to better expose hidden insights and relationships. We then train a parametric memory module to encode this knowledge in its parameters. Finally, we evaluate our approach by examining the balance between memorization and generalization, with the goal of developing a scalable system that effectively integrates new knowledge while maintaining strong performance on downstream tasks.

STRATA: An agentic training architecture for scientific task automation
Nicholas D. Abram
Mentor: Yisong Yue

AI agents offer a promising approach to scaling scientific data analysis by automating tasks that traditionally require substantial expert judgment. However, off-the-shelf agentic systems lack the domain expertise needed to perform the highly technical operations required by bespoke scientific data processing tasks. The performance of such agent systems can be undermined by high variability and irreproducible context. We introduce STRATA: Scientific Task, Rubric, and Agent Training Architecture, a framework for developing and evaluating reproducible scientific automation workflows via optimizing a rubric proxy that distills from expert examples and implements expert judgment in the form of executable and audible programs.

The architecture separates three roles. A Curator translates scientist-provided data, examples, and feedback into an adaptive evaluation rubric. An Implementor performs the scientific task by creating executable Python code, while an Evaluator assesses the Implementor's output against the rubric and produces grounded, actionable feedback. The Implementor then uses this feedback to generate a revised solution. The Curator receives the initial output, evaluator feedback, and revised output to determine whether the feedback produced meaningful improvement and to update the rubric for the next iteration. A private verification mechanism independently measures performance without exposing held-out evaluation information to the agents.

The architecture is designed to generalize across scientific domains and task types. By separating task execution, evaluation, and rubric development, the framework makes it possible to measure whether agent feedback produces meaningful improvement and whether the resulting guidance transfers to new tasks. The architecture therefore provides a general approach for building scientific agents that learn from expert judgment while preserving reproducibility and independent evaluation.

Discovering interpretable visual features in high-fidelity generative models
Trevor B. Chen
Mentors: Pietro Perona and Ilona Demler

Understanding and controlling the visual concepts represented inside generative models requires features that are both interpretable and causally testable. This project studies whether sparse feature methods can decompose a compact image representation into human-interpretable and editable factors. I use a pretrained FFHQ-128 diffusion autoencoder as an experimental substrate because its semantic representation is a single 512-dimensional vector with no spatial or token-position

dependence, unlike the spatially distributed residual streams of a DiT, while its decoder still produces realistic faces. I verified that the representation is deterministic, expanded the code corpus from 70,001 FFHQ examples to 232,771 FFHQ and CelebA examples, and trained TopK sparse autoencoders with different dictionary widths and active-feature budgets. Candidate features are evaluated by modifying their coefficients and passing the edited representation through the actual image generator, rather than by inspecting autoencoder reconstructions alone. This distinguishes features that correlate with an image property from features that causally control it. Initial scalar SAE experiments show that increasing data and capacity improves reconstruction, but feature usage and foreground/background entanglement remain important concerns. To test whether visual concepts occupy multidimensional subspaces, I added a Block-Sparse Featurizer comparison that selects coordinated blocks of latent directions. The goal is to discover stable edits to glasses, hair, face shape, lighting, and background, and to determine whether structured sparse representations provide cleaner image features than scalar SAEs.

Streaming state-space hybrid mesh recovery model
Xinran Xie
Mentors: Pietro Perona and Ilona Demler

We present Streaming State-Space Hybrid Mesh Recovery Model, an efficient streaming HMR with high reconstruction quality. Traditional video HMR is accurate but offline: the temporal module attends across the whole clip, which makes the clip a prerequisite for any single output frame and makes memory grow with the history kept. Hybrid state-space architectures remove such cost in language models, but a language model is already causal, so substituting a recurrent layer preserves what every token had. An offline mesh recovery model is bidirectional, and the same substitution removes the future along with the quadratic term. We resolve this causality mismatch by a retraining-free state-space replacement pipeline which enables streamability without significant loss in reconstruction quality, using only 8.5k frames from three benchmark datasets for fine-tuning. We achieve **(a). higher runtime and memory efficiency**. The converted model carries 1.87 MB of state per tracked person with no key-value cache, unchanged from 1,000 to 131,072 streamed frames and identical at every context window measured, against 25.2 times more state for an attention reference at a 960-frame context. Full replacement raises the processable clip length to 127k frames, by 8.3× the offline backbone's. We also achieve 1.39x and 1.15x speed up against the two streaming baseline models Human3R and Online HMR with higher performance. **(b). preserved accuracy on both in-domain and zero-shot tasks**. With fine-tuning, at 50% replacement, the hybrid beats the offline backbone it is derived from, and it beats SOTA streaming model baselines on all three evaluation sets. Furthermore, we propose not only a single model, but also a transferrable recipe for making a bidirectional mesh recovery architecture efficient, streamable, and high-quality.

Strategic risk seeking in games
Oscar Jenner
Mentors: Eric V. Mazumdar, Boris Velasevic, and Nicolas Lanzetti

Strategic risk seeking, that is, optimism about other players' strategies, can be beneficial both in coordination games, where such optimism is natural, and in highly competitive games, where disadvantaged players must rely on opponent mistakes. In the spirit of recent work on risk aversion in games, this report studies strategic risk seeking in finite games and in linear quadratic dynamic games, focusing on when the proposed equilibria are unique and tractable to compute. In the first part of the report, a new risk-seeking equilibrium is introduced, combining risk seeking with bounded rationality. Independently of the game payoffs, conditions on the players' levels of optimism and bounded rationality are given that guarantee uniqueness of the equilibrium as well as convergence of best responses to it. In the second part, a risk-seeking equilibrium for linear quadratic dynamic games is proposed, for which existence and uniqueness guarantees are given. Numerical experiments show that mutual optimism can ease competition, but can also lead to free-riding. The open-loop formulation is then considered, where monotonicity conditions are given, paving the way for model predictive control applications.

Neural operator splitting for Monte Carlo fluid simulation
Jianxiang Li
Mentors: Anima Anandkumar, Hong Chul Nam, and Xi Deng

Accurate fluid simulation around complex boundaries is computationally expensive, particularly for methods that use random sampling rather than geometry-conforming meshes. This project develops learned approximations of a velocity-based Monte Carlo fluid solver and investigates how the organization of learned updates affects error accumulation over long simulations. We introduce three complementary forms of neural operator splitting. The first learns separate updates for the solver's advection, diffusion, and pressure-correction stages. The second alternates models trained for successive positions in a prediction sequence, enabling later models to account for errors produced earlier. The third applies rotated or reflected versions of a learned update and then restores the result to its original orientation. The models are trained on solver-generated velocity fields and evaluated on flows with and without solid obstacles, including previously unseen airfoil shapes. Experiments indicate that all three strategies improve long-horizon prediction accuracy over repeatedly applying a single model, with their combination providing the strongest overall performance. Ongoing work extends the framework to three-dimensional and broader flow settings. Neural operator splitting thus offers a practical path toward more accurate long-range fluid prediction while preserving useful structure from the original solver.

GPU acceleration of TorchLean, a formally verified neural network training library in Lean 4
William H. Adkisson
Mentors: Anima Anandkumar and Robert Joseph George

TorchLean is a library for specifying, training, and formally verifying neural networks. TorchLean is written in Lean 4, an interactive theorem prover that serves as a proof assistant and functional programming language. Because every operation is implemented and checked within Lean, TorchLean avoids the semantic drift that arises when a verified model specification is converted to an external format for verification. This drift can let a proof about the converted model fail to reflect the model that actually runs. This safety came at a steep performance cost, however, since TorchLean's core operations originally ran entirely on CPU, hundreds of times slower than standard machine learning frameworks. This project migrated TorchLean's numerical core to the GPU: first with a custom CUDA matrix multiplication kernel, then with NVIDIA's cuBLAS library, then with targeted use of PyTorch's optimized kernels at the specific operations that profiling identified as dominating runtime. On a roughly 500 million parameter transformer model, these changes reduced TorchLean's slowdown relative to PyTorch from over 700 times to about 4.5 times, while preserving Lean's formal correctness guarantees. Future work will extend formal verification to the GPU operations themselves.

Verifying neural operators
Ryan L. Hsiang
Mentors: Anima Anandkumar and Robert Joseph George

Fourier neural operators (FNOs) learn mappings between functions but are evaluated numerically on finite grids, creating a gap between conventional grid-based verification and guarantees over continuous function spaces. We develop a sound interval bound propagation framework for certifying FNOs in the continuum limit. Our method applies sign-split interval propagation at a finite quadrature resolution and accounts for discretization error using analytically bounded Fourier kernels. To reduce the looseness of global estimates, we maintain cellwise enclosures of function values and spatial derivatives and use them to construct local quadrature corrections. These enclosures are propagated through spectral convolutions, pointwise lifting and projection layers, residual connections, and nonlinear activations. Bounds at grid nodes are then extended to the entire continuous domain using certified local Lipschitz constants, producing upper and lower-bound functions for every admissible input satisfying prescribed pointwise, supremum, and Lipschitz constraints. Numerical experiments on the FNO trained on the Poisson equation demonstrate the soundness and tightness of our approach. This framework provides a foundation for resolution-independent robustness verification of neural operators acting on continuous functions.

Length generalization in continuous diffusion language models with spectral representations and neural operators
Jucheng Shen
Mentors: Anima Anandkumar, Jiachen Yao, and Robert Joseph George

Continuous diffusion language models generate text by denoising continuous sequence representations, but generalizing beyond training lengths depends on both the representation and the denoising model. This project studies these two stages on six symbolic tasks. At the representation stage, four COSMOS autoencoder variants test how spectral losses and learned frequency attenuation affect fine-tuning. The attenuation variants adapt much faster, while the slope-loss-only variant catches up with longer training. This suggests that the early gap reflects optimization speed more than representational capacity. At the denoising stage, Transformer and Transformer–Fourier Neural Operator (TFNO) score models are compared in the COSMOS conditional-diffusion pipeline. TFNO does not outperform the Transformer on most tasks, and zero padding has mixed effects. An autoregressive control supports this general result, although prefix-TFNO substantially improves one multiplication task. Current experiments test whether attenuation provides a shorter fine-tuning path and whether spectral objectives conflict with reconstruction gradients. Follow-up diffusion experiments with token-aligned ELF embeddings will test whether the adaptation gap and padding behavior persist outside the compressed COSMOS latent space.

Transfer learning for AI-assisted mathematical discovery in combinatorial group theory
Caroline Zhang
Mentors: Anima Anandkumar and Robert Joseph George

AI is transforming mathematical research, with LLM and RL-based solvers finding proofs to previously unsolved problems. Motivated by these recent advances, and by how human mathematicians are able to transfer knowledge between problems, we investigated and enabled AI systems to learn general mathematical representations that can be applied across different problems. We focused our study on 24 families of problems in combinatorial group theory, including the notable Andrews–Curtis conjecture, an open problem for over 60 years. We developed a reinforcement learning (RL) framework and transfer learning pipeline, and we further trained a transformer-based foundation model capable of pretraining on all 24 families in the benchmark. Our results show that (1) transfer learning is effective across diverse group theory problems, yielding 17 robust source-to-target pairs with positive transfer after 1M training timesteps; (2) successful transfer can be attributed to both reusable learned representations and fine-tuning during training; and (3) pretraining on different source problems enables agents to solve complementary subsets of benchmark instances, resulting in greater dataset coverage than training from scratch alone. Together, these findings suggest that shared representations can emerge across problems in group theory and that training on one task improves performance on others, highlighting the potential of generalizable RL agents for mathematical solving and discovery.

Coarse-to-fine 3D MRI reconstruction via resolution-agnostic neural operators
Ruibo Wang
Mentors: Anima Anandkumar, Jiayun Wang, and Valentin Duruisseaux

Magnetic resonance imaging produces detailed images of the body but is slow to acquire, so a central problem is reconstructing high-quality images from undersampled measurements. We tackle this problem in three dimensions. Reconstructing a whole volume, rather than stacking independently reconstructed slices, keeps anatomy consistent through the slice direction, but it is memory-heavy, and training a network directly at clinical resolution is usually infeasible. Our method uses neural operators and is resolution-agnostic: we train it at low resolution and then reconstruct at higher resolution without retraining, a coarse-to-fine scheme that keeps training memory low. On thirty-three test volumes it outperforms a strong three-dimensional convolutional baseline by about three decibels in peak signal-to-noise ratio. The improvement holds on every volume, persists when measurement noise is added, and comes with roughly half the training memory. These results show that one resolution-agnostic operator can reconstruct volumetric scans more accurately than a fixed-resolution network while being cheaper to train. We are now extending the evaluation to full-resolution volumes and re-checking every number we report.