Ashish Rao
GitHub : github.com/aprao2002
Email : ashish.arartc@gmail.com
Mobile : 832-390-7091

Education

- **Stanford University** — Palo Alto, CA
  Combined Bachelors+Masters in Computer Science, GPA: 4.0 — *Sept. 2020 – June 2024*
  - **Coursework:** Reinforcement Learning, Natural Language Processing w/ Deep Learning, Parallel Computing, Compilers, Convex Optimization, Computer Vision w/ Deep Learning, Databases, Algorithms, and more.
  - **Technical Skills:** Languages – Python (7yrs), C++ (4yrs), Java (4yrs), C (3yrs), HTML/CSS/JS (1yr), basic R and Julia (< 1yr). Skilled in domains of ML, NLP, Reinforcement Learning and related frameworks & tools such as PyTorch, JAX, and Tensorflow.

Experience

- **NVIDIA (Deep Learning Frameworks - JAX)** — June 2024 - September 2024
  Software Engineering Intern
  - Developing a client API and distributed runtime to enable JAX users to write and run MPMD programs in a single-view controller execution environment.
  - Project is inspired by Google's Pathways architecture, and involves significant design and engineering work in the JAX and XLA codebases, as well as extensive usage of Ray.
- **HazyResearch Lab @ Stanford (Chris Re's Group)** — Jan 2024 - Present
  Student Researcher
  - Ran experiments with subquadratic models like Mamba and linear attention to investigate how non-causal and multi-pass processing can improve the performance of autoregressive sequence models: https://arxiv.org/abs/2407.05483v1.
  - Investigated the role of MLPs in the transformer architecture. Constructed synthetic problems to measure the performance of different MLP variants, produced theoretical characterizations of MLP capacity, and designed new MLP variants that showed improved results in synthetic benchmarks.
- **Google (Search Quality / Webanswers)** — Summer 2022 and 2023
  Software Engineering Intern
  - Led projects to detect and correct malformed search suggestions shown to users on the Things to Know feature from design to live experiments.
  - Developed high-precision ($> 90\%$) detectors for malformed search suggestions using PaLM and MUM large language models. Developed an approach using PaLM 2 to rewrite $O(100M)$ search suggestions with 95% precision and 80% recall.
  - Wrote performant online serving code and large-scale offline data processing pipelines in C++; configured TPU clusters for efficient bulk inference at > 2000 qps with LLMs; used Python to experiment with few-shot prompting and soft-prompt tuning; curated and analyzed training / evaluation datasets using SQL; evaluated final model performance using Google-internal search quality evaluation tools.
  - Trained Transformer-based retrieval models using self-supervised learning.
  - Curated diverse datasets to train / evaluate retrieval and malformedness detection / correction models. Gained significant experience in working with real-world data, techniques such as slice analysis, and formalizing ambiguous product goals into concrete engineering objectives and ML-based solutions.
- **Van Roy Lab @ Stanford** — Sept 2022 - Jan 2024
  Student Researcher
  - Developed a new framework and problem setting for agents that must learn from nonstationary streams of data: the computationally-constrained online continual learning setting (https://arxiv.org/abs/2307.04345).
  - As part of computationally-constrained OCL project, experimented with replay-based continual learning methods on variants of benchmarks such as PermutedMNIST and Gaussian Scheduled CIFAR-10. Also experimented with optimization-based meta-learning methods to automatically discover auxiliary objectives for training.
  - Designed novel prioritization schemes for Replay-based RL algorithms that leverage uncertainty estimates produced by neural network architectures such as Epistemic Neural Networks.
- **Dror Lab @ Stanford** — Jun. 2021 - Jun 2022
  Undergraduate Researcher
  - Applied Reinforcement Learning, SO(3) equivariant graph neural networks, and other ML techniques to optimize the drug discovery pipeline by developing new protein-ligand docking software. Extensively used Python, PyTorch, OpenAI Gym and domain-specific software such as RDKit and PyMol.

- Trained 3DCNN and equivariant graph neural network models to score poses generated by traditional protein-ligand docking software with high correlation (> 0.9 train, ≈ 0.7 test).
- Developed an ML based protein-ligand complex pose optimization technique that could improve the accuracy of high-error poses generated by traditional protein-ligand docking software by up to 35%.
- Developed an RL environment for protein-ligand complex pose optimization. Experimented with RL algorithms like PPO and SAC, and techniques such as meta-learning, genetic algorithms, metric learning, and imitation learning.

- **Stanford Student Robotics** Sept. 2020 - Apr. 2021

Robotics Team Member

- Developed a simulator for a project to build an autonomous boat to monitor coral reefs that modeled robot position, velocity, angular acceleration, and more, as well as the effects of ocean currents.
- Implemented motion planning algorithms for autonomous boat project including approaches based on sequential least squares programming, the A^* algorithm, and Voronoi diagrams.
- Wrote low-level controllers, telemetry radio code, and a testing framework to verify system correctness.
- Code written in Python and is available at: https://github.com/stanfordroboticsclub/boat-autonomy

Featured Projects

- **Prophet: Disaggregated LLM Inference with Optimized Request Schedulers** 2024

Course Project for CS 244B Distributed Systems

- Developed an optimized LLM serving system that minimizes time-to-first-token and time-per-output-token metrics. Disaggregated prefill and decode inference stages on different GPUs and designed new request schedulers to minimize head of line blocking and end-to-end latency.
- Evaluated system for LLAMA-3 8B inference on an 8x A10G instance. Demonstrated 28% lower median time-to-first-token and 78% lower 99th percentile time-per-output-token with custom disaggregation and scheduling over traditional methods.
- Details: https://tinyurl.com/prophet-llm. Code: https://github.com/cs244b-spr24/project.

- **Continual Learning as Computationally Constrained Reinforcement Learning** 2023

Research Project @ Stanford Intelligent Systems Laboratory (Van Roy lab)

- Co-authored a position paper synthesizing a formalism of continual learning as an instance of Reinforcement Learning. Reframed prior work in terms of this formalism, and explored several implications and research directions associated with our formalism.
- Implemented popular continual learning benchmarks and replay-based CL algorithms. Experimented with metalearning approaches, and experimentally characterized tradeoffs between stability and plasticity in deep neural network agents when applied to continual learning problems.
- Details: https://arxiv.org/pdf/2307.04345.pdf

- **Distributed Optimization with Block Diagonal Hessian Approximation** 2023

Course Project for CS 229 Machine Learning

- Developed a distributed neural network training algorithm called BlockSketchySGD which uses Hessian-vector products and a novel variant of the Randomized Nystrom approximation to precondition gradient updates using a block-diagonal approximation to the objective function Hessian.
- Algorithm can leverage block diagonal structure to compute Hessian approximation and gradient updates in a distributed manner across machines.
- Achieved higher test accuracy in lower wall-clock time than Adam on experiments training a CNN on CIFAR-10 and fine-tuning DistilBERT for sentiment classification.
- Details: https://tinyurl.com/block-sketchy-sgd. Code: https://github.com/Ashboy64/pytorch-hvp-optimizers.

- **Gaussian Process Policy Optimization** 2019 - 2020

Synopsys Science Fair, Intel ISEF

- Led award-winning project to develop a novel actor-critic, model-free Reinforcement Learning algorithm that uses a Gaussian Process and a novel surrogate objective to maximize reward in challenging environments. Details: arxiv.org/abs/2003.01074

Awards and Honors

- **Tau Beta Pi Engineering Honor Society Member**: Member of Stanford's premier Engineering honor society. Membership is granted based on class rank and community service.
- **Science Fair Awards**: **Grand Award at Intel International Science and Engineering Fair** (3rd Place in the Robotics and Intelligent Machines category), Best of Championship @ Synopsys Science Fair, Intel Excellence in Computer Science Award, Association of Computing Machinery award, Certificate of Congressional Recognition.
- **Hackathon Awards**: 1st Place @ SV Hacks for Android app allowing administrators to notify students if there's danger on campus. 2nd Place @ Los Altos Hacks 2 for an Android application which assists in speech memorization.