Allan Zhang

Last updated in March 2025

Los Angeles, CA allanzhang440@gmail.com https://giyushino.github.io giyushino

Education

University of California, Los Angeles *BS in Applied Mathematics* *Sept 2024 – June 2028*

- GPA: 4.00
- **Relevant Coursework:** Multivariable Calculus, Linear Algebra, Discrete Structures, Intro to C++ (Math 32A, Math 32B, Math 33A, Math 61A, CS 31)

Experience

Undergraduate Research Assistant | BigML *Los Angeles, CA*
UCLA *Nov 2024 – Present*

- Aided PhD student Siddharth Joshi in training and evaulating custom, lightweight VLMs on spatial reasoning tasks using mechanistic interpretability techniques (i.e. linear probing, ablation studies, and feature visualization)
- Wrote functions to create highly modular datasets, testing the effects of modality mismatch, spurious correlations, alignment between visual and textual context, and number of unique images per class on model's performance
- Modified training pipelines to train Jax transformers on Huggingface dataset, ran experiments regarding PiKE
- Attended weekly reading groups, keeping up to date with the recent advancements in the field of ML

Projects

CLIP *EvolutionSimulation*

- i did some fucking AWESOME SHIT test test test

TinyMathLLM *TinyMathLLM*

- Researched how datasets, model architecture, compute power, prompting, low-rank adapation, etc can be optimized to finetune LLMs. Demonstrated effectiveness of using LLMs to extract data from questions and using tools
- Finetuned TinyLlama (1.1B) on custom datasets entirely on RTX 4070, demonstrating effective math agents can be trained with limited resources. Despite training 0.0024% of params using LoRA and using 4-bit quantization, model experienced significant improvment in performance

Doodle Guesser *MyOwnDoogleGuesser*

- Fine-tuned OpenaAI CLIP-Vit-Large-Patch-14 and custom vision transformer on 6 different animal drawings. Collected and processed 1,000,000+ images of 6 classes from web to create custom dataset
- Using custom training + inference pipeline, accuracy increased from 54% to 87%, worst group improved from 39% to 76%. Custom model achieved 70% accuracy with only 200,000 training images
- Created GUI using Pygame to allow users to draw on a canvas. After drawing, screenshot was taken, processed, and fed into model, softmax taken of the outputted logits to predict the most likely class

Skills

Programming Languages and Frameworks: Python, PyTorch, NumPy, Matplotlib, TensorFlow, scikit-learn, OpenCV, Hugging Face, LaTeX, C++ (basic)

Languages: English, Korean