Kunj P. Shah

kunjcr2@gmail.com | +1 (628)-529-6990
LinkedIn | Github | Portfolio | San Francisco, CA

EDUCATION

San Francisco State University — San Francisco, California
B.S. in Computer Science
- GPA: 3.96/4.00, *Dean's List* — Expected Graduation 2027

EXPERIENCE

AI Agent Intern, Dreamable Inc., San Francisco, CA — *May 2025 – Aug 2025*
- Contributed with the team to finetune a Qwen-2.5-7B-param using Huggingface, PyTorch, Lambda (Cloud computing) on Q&A tasks and hosted on Cloud Run (Google Cloud Platform).
- Led Dataset curation using pandas, numpy and datasets library
- Used Low Rank Adaptation (LoRA) method from transformers library for cost efficient training
- Evaluated model and hyperparameters tuned to achieve very low valuation loss, tracked using wandb (model logging and experiment tracking).
- Additionally, Developed an AI-powered Outreach Agent using Langchain, Exa.ai along with OpenAI API Integration to automate messaging workflows. Currently used by 14+ interns to scale weekly outreach with minimal effort.

PROJECTS

Qwen-2.5-0.5B Finetune Github | Huggingface | Dockerhub
- Independently fine-tuned Qwen-2.5-0.5B using Hugging Face Transformers, PyTorch, LoRA, and DPO (post-training human alignment) on Google Colab A100 (GPU compute) for instruction-following tasks.
- Trained with bf16 (*lower memory usage*), gradient checkpointing, Flash Attention (*faster training*), and tf32 *(for memory efficiency and faster inference)*; experiments tracked in Weights & Biases *(experiment logging)*.
- Packaged an inference-ready Docker image powered by vLLM *(faster inference)*; artifacts published on DockerHub and mirrored on Hugging Face Hub *(deployment-ready)*

GatorGPT Github | Huggingface
- Engineered a 63M parameter transformer model using PyTorch and modern architecture components such as Grouped Query Attention, Rotary Positional Encodings, and SwiGLU MLP layers (for improved efficiency and contextual understanding), trained on the TinyStories dataset.
- Deployed and served using vLLM, with the complete model available on Hugging Face for one-click usage.
- Planned next phase involves fine-tuning on university-specific datasets using Direct Preference Optimization (DPO) and Reinforcement Learning *(for personalized alignment after supervised fine-tuning)*.

theHelper - AI Research Assistant Github
- • Engineered a **Retrieval-Augmented Generation (RAG)** system using **PyPDF2** *(PDF parser)*, **BERT** *(encoder-only transformer for embeddings)*, **Google Gemma** *(encoder-decoder model for Q&A)*, and **FAISS** *(vector database for semantic search)* — integrated seamlessly into a **Streamlit app** for real-time summarization and question answering.
- Designed an intuitive user interface for document upload, embedding generation, and response retrieval *(enabling semantic understanding of long-form PDFs in natural language)*.
- • Reduced manual review effort across academic and client documents by introducing context-aware retrieval and automated reasoning *(actively used by peers and family for coursework and professional summaries).*

And more on Github.