

# Kunj P. Shah

[kunjcr2@gmail.com](mailto:kunjcr2@gmail.com) | +1 (628)-529-6990

[LinkedIn](#) | [Github](#) | [Portfolio](#) | San Francisco, CA

## EDUCATION

San Jose State University

San Francisco, California

B.S. in Computer Science

- GPA: 3.96/4.00, *Dean's List*

Expected Graduation May 2027

## EXPERIENCE

**ML Engineering Intern, Routes Technologies, Remote, TX**

Oct 2025 – Present

- Working with the team to train and manage AI Models using Python, PyTorch and transformers; along with model tracking with wandb as well as model serving using endpoints on Azure ML Studio.
- Created a fully working Web Crawler using Scrapy and a scraper using BeautifulSoup4, providing company relevant data from open websites under proper ethics.
- Built a Flask-based Instagram Graph API integration using Python and PyDantic, with OAuth Authentication and automated hashtag/recipe detection.

**AI Engineer 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), LoRA (cost and memory efficient training) on Q&A tasks and hosted on Cloud Run (GCP).
- Led Dataset curation using pandas, numpy and datasets library
- 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.

## PROJECTS

**MedAssistGPT-401M** [Github](#) | [Huggingface](#) | [WandB](#)

- Built and pretrained a 401M-parameter GPT from scratch on 2M+ PubMed documents using PyTorch; implemented RoPE, Grouped Query Attention (GQA), SwiGLU, and RMSNorm as a modern transformer architecture.
- Optimized for A100 GPU with bf16, Flash Attention, gradient accumulation, and OneCycleLR scheduling; experiments tracked via Weights & Biases with automatic HuggingFace Hub checkpoint uploads.
- Developed a memory-efficient data pipeline using memory-mapped arrays (~0 RAM overhead) and custom XML-to-text cleaning for biomedical literature preprocessing.

**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*)

**Max – Personal Voice Assistant** [Github](#)

- Developed a voice activated AI Agent using Langchain, OpenAI, and SpeechRecognition to automate tasks along with hands-free interaction.
- Tools like, Web Search, Youtube Streaming, Emailing, File generation with wide range of extensions, knowledge based Retrieval Augmented Generation, controlling camera and much more are integrated.

And more on [Github](#).