Aryan Parab

310-869-1411 | amparab@usc.edu | Los Angeles, CA
linkedin.com/in/aryan-parab-0b44991b2 | github.com/aryanparab

Education

University of Southern California Los Angeles, CA
Master of Science in Computer Science – GPA: 4.0/4.0 *Jan. 2025 – May 2026*

- **Relevant Coursework:** Machine Learning, Artificial Intelligence, Data Science, Algorithms, Statistics
- **Returning to degree program:** After internship completion, graduating May 2026

Thadomal Shahani Engineering College Mumbai, India
Bachelor of Engineering in Computer Engineering *Aug. 2019 – May 2023*

Technical Skills

Languages: Python, Java, Scala, SQL
ML Frameworks: PyTorch, TensorFlow, Scikit-learn, Pandas, NumPy
ML Specializations: Supervised Learning, Unsupervised Learning, Recommendation Systems, NLP
ML Operations: End-to-End ML Pipelines, Model Training, Production Deployment, Scalable Systems
Software Engineering: Version Control (Git), Testing, Code Review, CI/CD
Tools: Spark, Docker, AWS, MongoDB, PostgreSQL

Experience

Deloitte Mumbai, India
Data Engineer / Machine Learning Engineer *Nov. 2023 – Dec. 2024*

- Built scalable ML pipelines processing 100K+ records weekly with automated validation achieving 99.2% accuracy in production deployment using version control and code review best practices
- Developed unsupervised ML system using K-Means clustering with scikit-learn organizing 50K+ documents reducing manual work by 60%, implementing end-to-end pipeline from training to deployment
- Optimized data processing performance by 40% collaborating with cross-functional teams translating business needs into ML solutions demonstrating strong communication and self-motivation

Projects

Scientific Paper Recommendation | *Python, PyTorch, BERT, Pandas, NumPy, Scikit-learn*

- Built end-to-end recommendation system using supervised learning (SVM) and unsupervised learning (LDA, K-Means clustering) with PyTorch BERT embeddings improving retrieval relevance by 22% over baseline
- Developed scalable ML pipeline processing 100K+ research papers using Pandas and NumPy for data processing, implementing model training workflows and production deployment infrastructure
- Experimented with new algorithms and techniques (BERT transformers, cosine similarity) to improve recommendation quality, applying software engineering best practices with Git version control and comprehensive testing

AI Vocal Coach | *React, Python, FastAPI, TensorFlow, Pandas, AWS*

- Designed and deployed end-to-end machine learning system with CNN audio classifier using TensorFlow serving 50+ users, building scalable ML infrastructure on AWS with model training and inference pipelines
- Developed data processing pipeline using Pandas and NumPy for audio feature extraction (MFCCs, spectrograms) optimizing inference latency by 30% through experimentation with model architectures
- Collaborated with cross-functional team translating user experience needs into ML solutions, implementing software engineering best practices including testing, code review, and continuous deployment

Multi agent orchestration game | *Python, OpenAI GPT-4, Anthropic Claude, LangChain, FAISS, FastAPI*

- Built scalable ML system with 10+ coordinating AI agents using Large Language Models (GPT-4, Claude APIs) optimizing performance from 2.1s to 1.2s through systematic experimentation with algorithms and architectures
- Developed end-to-end ML pipeline with RAG architecture using FAISS vector database for retrieval, implementing production deployment infrastructure with FastAPI backend and MongoDB persistent storage
- Applied ML operations best practices including version control with Git, automated testing, and code review demonstrating curiosity and self-motivation solving open-ended challenges in LLM systems

Imitation Learning for Robotic Control | *Python, PyTorch, NumPy, MuJoCo, OpenAI Gym*

- Implemented supervised learning (behavioral cloning) and reinforcement learning (DAgger) algorithms using PyTorch achieving 99.9% of expert performance (3714.5 vs 3717.5 return) on robotic locomotion tasks
- Built end-to-end ML training pipeline processing trajectory data with NumPy demonstrating practical experience with ML operations including hyperparameter tuning, model evaluation, and iterative data collection
- Experimented with training techniques improving performance +316.7% over baseline through systematic algorithm iteration, applying data science methods analyzing compounding errors and distributional shift