

# Ritam Chakraborty

408-529-4668 • [ritam.chakraborty@gmail.com](mailto:ritam.chakraborty@gmail.com) • [ritamchakraborty.com](http://ritamchakraborty.com) • [linkedin.com/in/ritam2006](https://linkedin.com/in/ritam2006) • San Jose, CA

## EDUCATION

### University of California, Berkeley

Bachelor of Arts in Data Science

GPA: 3.96 • Berkeley, CA

Expected Graduation: 05/2027

- **Relevant Coursework:** Data Structures (A), Principles and Techniques of Data Science (A), Linear Algebra and Differential Equations (A), Discrete Mathematics and Probability Theory, Multivariable Calculus

## EXPERIENCE

### Founding Engineer

SomeIdea AI

Berkeley, CA

03/2025 – Present

- Promoted from Intern to Founding Engineer in recognition of exceptional performance at a startup building AI-driven solutions to streamline research and analysis workflows for buy-side investment professionals
- Spearheaded development of a transcript analysis tool that leveraged natural language processing (NLP) to condense 100+ pages of financial documents into concise investor summaries, distributed to 50+ paid users
- Designed schema to standardize financial document ingestion, reducing downstream data-cleaning by 35%
- Engineered pipelines to migrate 100+ documents from Google Cloud to AWS Relational Database Service

### Backend Developer

Berkeleytime

Berkeley, CA

09/2025 – Present

- Deployed features for UC Berkeley's premier course discovery and planning platform that serves 10,000+ users
- Enhanced GraphQL schemas and operations to enable monitoring of class capacity during enrollment periods
- Incorporated jobs to monitor enrollment data and alert users when tracked classes reached capacity thresholds
- Collaborated with frontend, product, and design teams to develop APIs supporting new course tracking tools

### Machine Learning Consultant

Balnce AI

Remote

01/2025 – 05/2025

- Fine-tuned LLMs for entity extraction, achieving 25% accuracy improvement over baseline Qwen performance
- Created data quality monitoring framework with anomaly detection to flag inconsistencies in training data
- Developed data generation and validation pipelines in Python producing 1,000+ synthetic training examples
- Automated testing framework for 50+ hyperparameter combinations, reducing manual tuning time by over 80%

### Frontend Developer Intern

UpUnikSelf

Remote

01/2025 – 03/2025

- Created 50+ interactive React.js components featuring state management, API integrations, and animations
- Optimized API response times by 40% through implementation of request batching and caching strategies
- Integrated usage analytics to track engagement, generating insights that informed product design improvements
- Designed responsive layouts for 30+ pages using various layouts, improving user experience across devices

## PROJECTS

### Portfolio Management Risk Estimator ↗ • Java, Spring, SQL, PostgreSQL, TypeScript, React.js, Next.js

- Implemented scheduled jobs to fetch and store market data for top 50 companies into a PostgreSQL database
- Computed portfolio risk metrics including Value at Risk using Monte Carlo simulations and Sharpe coefficients
- Enforced PostgreSQL connection pooling, reducing data retrieval latency by 40% for real-time risk calculations
- Created Next.js frontend for portfolio creation and risk visualization, integrated with Spring Boot REST API

### Chess Tablebase Generator ↗ • C++

- Programmed a retrograde analysis engine to solve 3-6 piece chess endgames across over 3 trillion positions
- Engineered a custom state encoding scheme with symmetry pruning and Zobrist hashing to minimize storage
- Incorporated probabilistic inference and value propagation across game trees to determine position evaluation
- Built multi-threaded architecture with work queue distribution reduce processing time from weeks to days

### Image Recognition for Medicine Package Quality Control ↗ • Python, Pandas, NumPy, TensorFlow

- Applied edge detection algorithms and transfer learning techniques to detect defects in medicine packaging
- Implemented dual-perspective system analyzing side and top-view images to classify package defects
- Compared performance between trained neural networks against traditional machine learning classifiers
- Published methodology and implementation recommendations in an academic journal as the sole author

## SKILLS

**Languages:** Python, Java, C++, Go, SQL, TypeScript, JavaScript, HTML, CSS

**Frameworks:** Flask, FastAPI, Spring, Spring Boot, Node.js, Express.js, React.js, Next.js, Vue.js

**Libraries:** Pandas, NumPy, TensorFlow, PyTorch, Apache Commons, Axios, Tailwind CSS

**Other Tools:** Docker, Git, Linux, PostgreSQL, GraphQL, Jupyter Notebook, Firebase, Supabase