

Daniel Lee

703-470-5155 | danielyedarmlee@gmail.com | linkedin.com/in/daniel-lee-2b1474326 | github.com/danielylee2006

EDUCATION

Yale University

B.S. in Computer Science

New Haven, CT

Expected May 2028

- **Relevant Coursework:** Data Structures and Algorithms, Software Engineering (in progress), Full Stack Web Development (in progress), Computer Systems (in progress – operating systems, computer architecture, memory management, concurrency), Data Analysis, Linear Algebra, Multivariable Calculus
- **Organizations:** Yale Computer Society, Yale Varsity Fencing Team, Yale Entrepreneurial Society

EXPERIENCE

Aria

Software Engineer & Co-founder

Mar. 2026 – Present

New Haven, CT

- Built an AI-powered iOS app for college students using Python, TypeScript, PostgreSQL and Celery, **unifying classes, deadlines and calendars into a prioritized timeline and saving ~30 minutes/day of context-switching**
- Developed 4 LLM-powered pipelines with the Anthropic Claude API for document parsing, briefings, email classification and task management, **reducing projected AI inference cost by 55%** through model-tier routing
- Engineered asynchronous backend services and integrations across Canvas, Google Calendar, Gmail and Twilio, implementing OAuth 2.0, Google Cloud Pub/Sub, Celery/Redis task processing and Docker Compose

Mylon AI

Software Engineering Intern

May 2026 – Aug. 2026

Remote

- Deployed a RAG-based policy Q&A platform for university staff using LlamaIndex and pgvector on a 3-person engineering team, **reducing policy lookup time by 94%** through citation-grounded retrieval
- Automated the document ingestion pipeline converting PDF, DOCX and HTML handbooks into 1024-dimension Voyage AI embeddings, implementing parsing, chunking and per-chunk embedding-version tracking with LlamaIndex
- **Reduced hallucination risk by 88%** by implementing retrieval-aware fallback behavior that declines questions unsupported by institutional documents

Arbol Inc.

Software Engineering Intern

May 2025 – Aug. 2025

New York, NY

- Built an automated insurance contract extraction tool using AWS Textract OCR and a LangChain/OpenAI extraction pipeline, **cutting per-contract review time 40% against manual entry**
- Tuned the extraction prompt against noisy, unstructured OCR output with fuzzy field-name matching and explicit anti-hallucination fallbacks, **raising field extraction accuracy 30% over manual entry**
- Built an interactive React review interface with row-level locking, converting unstructured extraction results into a validated, pre-populated dataset reviewers can rapidly correct and approve

PROJECTS

LabLink | *Next.js, TypeScript, FastAPI, PostgreSQL/pgvector, Supabase* | lablinkusa.org

- Built and deployed a full-stack donation platform for surplus lab equipment, **redistributing \$40,000+ across 50+ fulfilled donations**, replacing a manual matching process that took days to a week per match
- Shipped an advisory ranking system that shortlists which institution should receive a donated item, **cutting median admin recipient-selection time by 65%**, using pgvector hybrid retrieval scored on equipment fit, location, urgency and wait time
- Engineered a CI/CD pipeline that gates every merge, **cutting pre-deploy verification time by 80%**, using PostgreSQL and pgvector integration tests with Playwright end-to-end coverage

AI Fencing Referee | *Python, PyTorch, OpenCV, scikit-learn, NumPy* | [GitHub](https://github.com)

- Trained PyTorch models under a strict athlete-and-event-disjoint evaluation protocol, **reaching 72% right-of-way accuracy, exposing a 12-point accuracy inflation when the same athlete appears in both training and test**
- Developed an end-to-end computer vision and data preprocessing pipeline in Python to ingest competition footage, detect touch events and segment temporal exchanges, **auto-generating weak-supervision labels for 9,000+ clips and cutting manual annotation effort by 92%**
- Designed a leakage-resistant ML data pipeline with schema validation, feature isolation and negative controls to block scoreboard-derived information from model inputs, **reaching 96% agreement with human-verified labels**

TECHNICAL SKILLS

Languages: Python, JavaScript, TypeScript, SQL, HTML/CSS, C/C++, R

Frameworks and Libraries: React, Next.js, FastAPI, Node.js, Celery, React Native, Playwright

AI/ML: PyTorch, scikit-learn, OpenCV, NumPy, LlamaIndex, LangChain

Databases and Infrastructure: PostgreSQL, pgvector, Supabase, Redis, Docker, Vercel, Railway, Google Cloud Platform

Developer Tools: Git, GitHub Actions, VS Code, Xcode, Expo, Claude Code, Codex