Nelson Kang

+1 (613) 854-3568 Nelsonkang7@gmail.com
linkedin.com/in/nelson-kang-ekanl github.com/Ollienel777

Technical Skills

Languages: Python, C#, C, SQL, TypeScript, JavaScript, HTML/CSS
Frameworks: FastAPI, Next.js, .NET, PyTorch, Pandas, NumPy, Celery, SQLAlchemy
Infrastructure: PostgreSQL, SQL Server, SSMS, Supabase, Redis, Modal, Cloudflare R2, Git, Linux, Power BI
Tools: Jira, FFmpeg, OpenCV, YOLOv8, Selenium, Alembic

Experience

Ericsson Canada Inc. **Jun 2024 – Aug 2024**
Business Intelligence Co-op *Ottawa, ON*

- Built 4 ETL pipelines using C# and SQL Server to extract Jira data from JSON payloads, transform custom field mappings, and load into reporting tables; used temporal tables for full change-history tracking.
- Audited and catalogued all existing team pipelines, identifying and documenting faulty custom-field mappings producing NULL data; created a reference spreadsheet that became the team's pipeline debugging resource.
- Developed and maintained Power BI dashboards for trouble report visibility, supporting stakeholder reporting and data-driven decision-making across the BI team.
- Led a technical workshop on building LEGO Mindstorms autonomous vehicles, implementing image-based sign recognition with tuned confidence thresholds.

Projects

ClipFarm — Automated Volleyball Highlight Generator **2026 – Present**
Python, FastAPI, Celery, Next.js, TypeScript, YOLOv8-pose, PostgreSQL, Redis, Cloudflare R2, Modal

- Built a full-stack video processing pipeline: users upload game film, YOLOv8-pose extracts player skeletons frame-by-frame, a rule-based action classifier scores spikes, serves, digs, sets, and blocks, and FFmpeg cuts and re-encodes clips to H.264, served via presigned Cloudflare R2 URLs.
- Added audio energy analysis (RMS on extracted PCM) to weight detection confidence by crowd noise, reducing false-positive clips by ~20%.
- Designed the backend for async scale: FastAPI handles auth-gated REST endpoints, Celery + Redis manages video processing jobs, Modal provides GPU offload for YOLOv8 inference. Implemented Supabase Auth (ES256 JWKS verification), per-user data isolation, and Row-Level Security.
- Built the frontend in Next.js + TypeScript with a filterable clip grid, inline multi-label correction UI for future model training, clip trimming with background re-cut via Celery, bulk operations, and lazy-loaded thumbnails.

ChessAI — Neural Network Chess Engine **Nov 2025**
Python, PyTorch, Modal

- Built a neural-network-guided chess engine trained on millions of Stockfish-labeled positions using distributed GPU compute on Modal.
- Designed a scalable ML pipeline for ingesting and labeling large Lichess datasets, handling data cleaning, feature extraction, and batch processing.
- Integrated classical alpha-beta search with deep neural evaluation for real-time gameplay, balancing search depth with inference latency.

Multi-Factor Robo-Advisor — 1st Place, Market Beat Competition **Nov 2025**
Python, Pandas, NumPy, yfinance

- Designed and implemented a multi-factor portfolio construction pipeline applying momentum, volatility, correlation, and trend signals inspired by the Ranked Asset Allocation Model (RAAM).
- Built end-to-end automation: data ingestion, liquidity screening, factor ranking, sector-cap enforcement, and Sharpe-ratio optimization under real-world constraints (transaction fees, FX conversion, position limits).
- Optimized a $1M CAD portfolio across U.S. and Canadian equities; achieved **5.7% return** vs. 3.7% benchmark, placing **1st out of 18 teams**.

Education

University of Waterloo
BMath, Computing and Financial Management (CS + Finance) *Expected 2029*