Alexander Kozik

215-264-2104 | alex.kozik@yahoo.com | linkedin.com/in/alex-kozik/ | alexkozik.com | github.com/LambdaAK | Ithaca, NY

Education

Cornell University — *B.A. Honors Computer Science, B.A. Mathematics* Aug 2022 – May 2026

- **Relevant Coursework:** Machine Learning, Deep Learning, Analysis of Algorithms, OOP and Data Structures (Honors), Systems Organization, Systems Programming, Functional Programming, Programming Languages and Logics, Discrete Structures, Probability Theory, Advanced Linear Algebra, Multivariable Calculus, Real Analysis, Macroeconomics
- **GPA: 4.1/4.0 (Top 1 %)** Dean's List, Focus on Machine Learning and Software Engineering

Professional Experience

Salesforce — *Software Engineer Intern* May 2025 - Aug 2025

- Develop custom virtual formula field functionality in Salesforce Data Cloud, empowering customers to create dynamic data transformations without code and enabling self-service analytics capabilities.
- Build scalable backend services using **Java** and **Apache Spark** for formula computations on distributed metadata services, enabling processing that scales automatically with customer data volume.
- Implement **REST API** endpoints with robust error handling and validation, providing developers with reliable programmatic access to formula field operations and reducing integration complexity.
- Create intuitive **React** and **TypeScript** frontend interface that simplifies virtual formula field creation and data exploration, dramatically improving user experience and accelerating customer onboarding.

CMSX — *Frontend Lead, Vice President, Fullstack Engineer* Aug 2023 – Present

- Maintain Cornell's course management system, used by **10,000+ students** in **100,000+ line** codebase.
- Lead frontend team by designing project architectures, reviewing PRs, and interviewing candidates
- Architected and built the student-facing frontend in **React**/TS, implementing a component architecture and state management to reduce loading times by 90% from the previous implementation.
- Implemented **APIs** with serializable data structures, enabling secure communication between frontend and backend while improving developer velocity through enhanced code modularity and reusability.

Carnegie Mellon University — *Software Engineer and Research Intern* May 2024 - Aug 2024

- Worked on the interpreter for SASyLF, a language for writing proofs, implementing core features in Java
- Implemented a polymorphic type system for SASyLF, similar to Java generics, enabling type-safe code reuse and significantly reducing duplication in proofs by allowing abstraction over multiple data types.
- Created a modular testing framework using **Python** to automate integration tests, ensuring reliability.

Cornell Generative AI — *AI/ML Engineer & Project Manager* Feb 2025 - Present

- Developed AI agent for QuickFi, a commercial lending technology company, automating insurance certificate validation and compliance verification processes, reducing operational costs.
- Built frontend and vector database for ClassGPT, enabling professors to create AI tutors for classes

Teaching Experience

CS 3780/5780: Introduction to Machine Learning — *Teaching Assistant* Jan 2025 – Present

- **Topics Covered:** Clustering algorithms, PCA, MLE/MAP, regression, optimization algorithms, SVM, kernelization, ERM, decision trees, ensemble methods, neural networks, CNNs, generative AI, and AI ethics.
- Held office hours helping students with homework assignments and programming projects, graded assignments and exams, and proctored exams.

CS 4820/5820: Introduction to Analysis of Algorithms — *Teaching Assistant* Aug 2024 – Dec 2024

- **Topics Covered:** Greedy algorithms, dynamic programming, network flow, divide and conquer algorithms, NP hardness, approximation algorithms, computability, and cryptography.

- Held office hours helping students with homework assignments and programming projects, graded assignments and exams, and proctored exams.

CS 3110: Data Structures and Functional Programming — *Teaching Assistant* Aug 2023 – May 2024

- **Topics Covered:** OCaml programming, polymorphism, models of evaluation, functors, mutability, logic, concurrency, data structures, lambda calculus.
- Awarded the CS course staff award for outstanding service to students.
- Held office hours helping students with homework assignments and programming projects, lead weekly recitations, graded assignments and exams, and proctored exams.

Personal Projects

C-Torch - Artificial Intelligence/Machine Learning Library | *C++*

- Developed a high-performance **math library** supporting numerical methods, calculus, and linear algebra.
- Implemented 15+ ML models, such as neural networks, achieving 99%+ accuracy on classification tasks.
- Implemented DQN and Policy Gradient agents achieving **97%+ win rates** in autonomous gameplay.
- Achieved 26x performance speedup through compiler optimizations and parallelization techniques.

LambdaScript - Custom Programming Language | *TypeScript*, *OCaml*, *Jest*, *OUnit*

- Built interpreter for custom programming language, achieving **50% speed improvement** over Python.
- Designed parser and type inference algorithm to validate programs and provide developer feedback.
- Implemented core language features including pattern matching, functions, and polymorphism.

HabitStack - Habit Tracking Application | *React*, *TypeScript*, *JavaScript*, *SASS*, *Express.js*, *Firebase*, *Vite*

- Built full-stack habit tracking application with **OAuth2**, calendar interface, and goal management.
- Developed **TypeScript APIs** and **Express.js backend** integrated with **Firebase** for seamless persistence.
- Implemented live chat functionality using database connections, enabling low-latency communication.

CritterWorld - Evolving Artificial Life Simulator | *Java*, *JavaFX*, *SceneBuilder*, *Gradle*, *JUnit*

- Built **JavaFX** GUI app to visualize real-time interactions and evolution of virtual organisms.
- Implemented custom programming language with parser and interpreter to model organism behavior.
- Developed pathfinding algorithms including **A*** and **Dijkstra's** for organism navigation.

Visual Attention Image Captioning - Show, Attend, Tell Implementation | *PyTorch*, *Python*

- Implemented attention-based image captioning model with **superior METEOR scores** vs. original paper.
- Built soft and hard attention mechanisms with **REINFORCE** and backpropagation for training.
- Developed encoder-decoder architecture using **ResNet-50** and **LSTM** with attention visualization.

AlgoSandbox - Interactive Algorithm Visualizer | *React*, *TypeScript*, *Vite*, *CSS*, *JavaScript*, *Netlify*

- Built interactive platform for visualizing classic algorithms and data structures with real-time animations.

Technical Skills

Languages: Python, JavaScript, TypeScript, Java, C/C++, Go, Rust, SQL, HTML/CSS, OCaml
Frontend: React, Next.js, Redux, Material-UI, TailwindCSS, SASS, JavaFX, Electron.js, JSP
Backend & Databases: Express.js, Flask, Firebase, Apache Spark, REST APIs, Node.js
ML Frameworks & Libraries: PyTorch, TensorFlow, NumPy, Pandas, Scikit-learn, OpenCV, Matplotlib, Seaborn
DevOps & Tools: AWS, Docker, Kubernetes, Git, Linux, Gradle, Maven, JUnit, Jest, Vite
Machine Learning: Neural Networks, Computer Vision, NLP, Reinforcement Learning, Deep Learning