

Estella Xu

US Citizen • (425) 770-4175 • estellaxu2025@u.northwestern.edu • [LinkedIn](#) • [Portfolio](#)

EDUCATION

Northwestern University

M.S. in Computer Science | GPA 4.0/4.0

Evanston, IL

Expected June 2026

B.A. in Mathematics and Computer Science | GPA: 3.7/4.0

June 2025

Relevant Courses: Database Systems, Scalable Software and Cloud, AI & ML, Data Science Pipeline, Distributed Systems, Data Structures, Algorithms, OS, Parallel Computing, Networking

SKILLS

Programming: Python, C/C++, SQL, JavaScript, R, Pandas, NumPy, TensorFlow, scikit-learn
Tools: AWS, Azure, Kubernetes, Docker, Airflow, Spark, Elasticsearch, Git, Linux, Jira
Languages: Fluent in English and Mandarin

WORK EXPERIENCE

360 Privacy

Remote

Software Engineer Intern

September - November 2025

- Built a **scalable data pipeline** to parse, clean, and index PII data from large dark web leaks (20GB+ files).
- Developed a **dynamic parser generator** using LangGraph with AI-assisted parser creation, automated testing, and human-in-the-loop validation to **handle malformed data**.
- Optimized I/O performance via batching & memory management, enabling **stable throughput on large inputs**.
- Embedded 100B+ records into Pinecone **VectorDB** for high-speed **fuzzy matching** and **semantic search**.
- Integrated the vector index with a Neo4j **graph DB**, linking client entities and relationships to enable pivot search.

New York Metropolitan Transportation Authority

New York, NY

Data Engineer Intern

June - August 2025

- Built and maintained **ETL pipelines** for the MTA's **datalake of 890+ Delta tables** on Kubernetes and Docker.
- Collaborated in **Agile development** with cross-functional teams via Jira; automated reports using the Jira API.
- Reduced memory usage by 50%** for Kubernetes pods and **runtime by 30%** through optimizing DB-API usage.
- Reduced memory usage by 25%** by modifying the **open source delta-rs** library's buffer flushing mechanisms.
- Automated 800+ DAGs** by extending the MTA's **custom Airflow library** for ingestion into Azure Blob.
- Repaired and optimized **Airflow DAGs**, deploying fixes via **GitLab CI/CD** to ensure reliable automation.

PrizeSole

Remote

Full Stack Software Engineer Intern

June - August 2024

- Implemented **full-stack features** for an e-commerce platform with discount and raffle mechanics.
- Developed a **front-end in Nuxt.js** with TailwindCSS and PrimeVue to enhance UI/UX.
- Led development** for the checkout and sign-up pages. Integrated the **Stripe API** for secure transactions.
- Built **AWS** serverless architecture with Lambda and API Gateway to manage user sign-ups and orders, ensuring **scalable, efficient data management**.

RESEARCH ([LINK](#))

Northwestern University | Distributed Databases on Heterogeneous Clusters ([Link](#))

September 2025 - Present

- Designed and implemented **hardware-conscious scheduling and query execution** for a distributed database on a heterogeneous cluster (12 Raspberry Pis, 2 Rock Pros, 1 server), improving **runtime, cost, and power usage**.
- Extended Spark's scheduler** for heterogeneous environments, dynamically accounting for **node-specific memory and compute capabilities**.
- Deployed and optimized the cluster using PXE boot to test OS and JVM configurations, and automated **TPC-H benchmarking** and management with **Ansible** for scalable performance evaluation.

Northwestern University | Join Implementations on Distributed Databases ([Link](#))

March - June 2025

- Reduced query execution time by 70%** by implementing join coordination in a **distributed DBMS**.
- Evaluated **scalability across 1-16 nodes** using **MPI** to implement joins on Northwestern's HPC cluster (Quest).
- Decreased runtime by 55%** by optimizing joins with **multiprocessing** and MPI non-blocking operations.

TECHNICAL PROJECTS ([LINK](#))

Fridge Application in Amazon Web Services ([Link](#)) — Python, JavaScript, AWS, RESTful APIs

- Designed and deployed a **full-stack application** enabling users to track fridge inventory and generate grocery lists from ingredients parsed in user-uploaded recipes.
- Used **RESTful APIs** with **RDS** for user data management and **S3** for PDF recipe storage; automated grocery list generation using **Lambda**. Added **user authentication** for secure access.
- Developed client-side logic in Python and backend Lambda functions to build a serverless, event-driven architecture, **reducing infrastructure overhead and improving scalability**.