

# Yuxin 'Chloe' Chen

## EDUCATION

University of Minnesota Minneapolis, MN

Bachelor of Science, Computer Science **GPA: 3.97/4.00**

Expected graduation date: May 2026

Relevant Coursework: Data Structures and Algorithms, Distributed Systems, Operating Systems, Database Management Systems, Software Engineering Principles, Cloud Computing, Machine Learning Systems

## SKILLS

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

Frontend: **React, Redux, D3.js**, Material UI, **NextJS**, Tailwind CSS

Backend: **Node.js, Express, Spring Boot, Django, RESTful API, GraphQL**

Database: **MongoDB, MySQL, PostgreSQL, Redis, Elasticsearch**

DevOps & Tools: **Git, Linux, Docker, Kubernetes, CI/CD, AWS, Azure, Jenkins**

ML & AI: **PyTorch, HuggingFace, TensorFlow, scikit-learn, LangChain, OpenAI SDK**

## PROFESSIONAL EXPERIENCE

**Snarkify, Inc. | Palo Alto, CA**

Software Engineering Intern | Jun 2024 – Aug 2024

- Architected zero-knowledge proof solutions processing **100K+ transactions**, achieving **95% efficiency**
- Engineered caching system with **Redis/LRU**, achieving **80% cache hit rate** and **25% latency reduction**
- Developed monitoring infrastructure using **Prometheus/Grafana**, automating error detection and reducing debug time.
- Redesigned authentication system implementing **JWT/OAuth 2.0/RBAC** with comprehensive **JUnit/Jest** testing

**Beijing Jinxinxiutu Technology | Beijing, China**

Software Development Intern | Jul 2023 – Aug 2023

- Built microservices for property management platform serving **10K+ users**, improving response time by **35%**
- Optimized **MongoDB** queries with indexing/connection pooling, reducing API latency by **40%** supporting **500+ users**
- Deployed containerized services on **Docker/Kubernetes**, implementing **CI/CD pipelines** reducing deployment time
- Collaborated using **Agile methodologies**, participating in standups/sprints improving team velocity by **25%**

**University of Minnesota | Minneapolis, MN**

Teaching Assistant - Introduction to Algorithms, Data Structures, and Program Development | Jan 2025 – Present

- Lead labs teaching **data structures/algorithms** in **Python/Java** for **40+ students**, improving comprehension by **30%**
- Created solution guides covering **binary trees/graph algorithms/dynamic programming**, receiving **4.8/5.0** rating

Research Assistant – GNN-101 Supervisor: Qianwen Wang | Oct 2024 – Present

- Developed educational visualization tool illustrating **GNN layer-by-layer** learning with interpretable views
- Implemented interface using **Next.js/React/D3.js/ONNX**, enabling real-time graph network structure exploration
- Built interactive node-link diagrams and matrix heatmaps highlighting hierarchical details through flow charts

## PROJECT EXPERIENCE

**Interactive LLM Summarization Tool | React, Node.js, MongoDB, Google Gemini AI, D3.js**

- Engineered AI document summarization platform reducing review time by **20%**, integrating **Google Gemini API**
- Implemented version-controlled document tree visualization using **react-d3-tree/Redux**, enabling content exploration
- Designed bidirectional text-highlighting with **PDF.js** rendering, achieving **sub-100ms** response for location navigation
- Deployed on **AWS EKS** using **Docker/Kubernetes**, implementing autoscaling handling **500+ concurrent users**

**Advanced NLP & Machine Learning Projects | PyTorch, HuggingFace, TensorFlow, AWS SageMaker**

- Designed **text classifier** in **PyTorch** achieving **92% accuracy**, optimizing hyperparameters with **Optuna**
- Fine-tuned **DistilBERT** on **SST-2 dataset** using **HuggingFace Transformers**, achieving **94.3% accuracy**
- Built text generation evaluation framework benchmarking **LLM sampling strategies** using **BLEU/ROUGE metrics**
- Deployed models as **Docker/Flask**, implementing **CI/CD pipeline** with **GitHub Actions** for **AWS EC2** deployment

**Distributed Task Scheduling System | Spring Boot, React, PostgreSQL, Redis, Kafka | Jan 2024 – Mar 2024**

- Architected task management app supporting **20+ users** and **1,000+ concurrent tasks** using **Spring Boot/React**
- Designed data layer with **PostgreSQL/Redis**, implementing **database sharding** reducing query latency by **60%**
- Implemented notifications using **Kafka**, enabling instant updates while maintaining distributed system consistency