CHEN GONG

(919)-619-9751 | gongchen07@outlook.com | LinkedIn | GitHub

SUMMARY

Software engineer specializing in AI infrastructure and scalable backend systems. Expert in building high-concurrency platforms, with a track record of architecting MCP-based agent workflows that handle 5,000 RPM and TB-scale data. Proven ability to optimize distributed ML(Machine Learning) inference latency from seconds to sub-450m. Strong in Python, TypeScript and C++, with hands-on experience in Docker, CI/CD, and modern web architectures.

EDUCATION

Duke University - *M.S. in Electrical and Computing Engineering (CS Major),* GPA: 3.7 | Aug 2024 - May 2026
The University of Manchester – *B.S.in Electrical and Electronic Engineering*, GPA: 3.9 | Sept 2021 - Jun 2024

EXPERIENCE

Scam AI | Berkeley, CA **Software Engineer (AI Infrastructure) |** May 2025 – Sep 2025

Scam AI ($2.5M Seed) is an AI security startup building next-gen Trust & Safety infrastructure, awarded 1st Place at HYSTA Annual Conference and backed by Berkeley SkyDeck.

- Built an **AI-driven Risk Management platform** detecting multimodal threats (Deepfake, document forgery) for **Fintech and Content Moderation** clients, **protecting $5M+ in potential transaction fraud** and supporting **5,000 RPM** and **10K+ daily risk assessments**.
- Enabled **real-time fraud intervention across multiple business scenarios** (payments, messaging security, content moderation), scaling system to handle **TB-scale monthly data** via S3 pre-signed URL ingestion (100MB+ per request).
- Engineered **a TypeScript-based MCP (Model Context Protocol) orchestration layer** to map 12+ agentic tools to internal REST APIs; **decoupled agent workflows from underlying ML services** via a zero-auth-duplication model, improving system modularity and scalability.
- Optimized **end-to-end detection latency to 300-450ms** through a **dual-path inference architecture**: implemented synchronous HTTP for fast-path checks and **Redis-backed (BRPOP) asynchronous task queues** with Python ML workers for compute-intensive media analysis.
- Designed **a Facade-based API Gateway** to unify 8+ fragmented ML detection pipelines into production-ready services, reducing integration overhead for new security features and enhancing platform extensibility.

Omexom | Manchester, UK **Electrical Engineer** | Jun 2023 - Sept 2023

- Developed automated reporting and alerting tools for transmission power systems, incorporating data validation and visualization to support operational decision-making.

PROJECTS

Machine Learning for Antenna Design

- Built machine learning models (KNN, Neural Networks, LASSO) to predict antenna return loss using **5GB+ simulation data**, achieving **92% prediction accuracy.**
- Performed model comparison and hyperparameter tuning to optimize performance across multiple algorithms
- Reduced model inference latency by **~30%** through feature selection and model optimization.
- Developed reusable training and evaluation workflows to support experimentation and result reproducibility.

Distributed Service Platform

- Designed and built a full-stack distributed service platform integrating task management, real-time communication, and data-driven workflows for individual users and small teams.
- Developed backend services using **Node.js, Django, and PostgreSQL**, **C++ proxy**, implementing **Prisma ORM** for type-safe database access and complex schema management.
- Orchestrated containerized deployment using **Docker** and automated **CI/CD pipelines**, ensuring environment consistency and 99.9% uptime for daily active users.
- Built a modular **React/TypeScript** frontend with state management, ensuring a seamless user experience across high-frequency real-time update scenarios.

TECHNICAL SKILLS

Languages: Python, TypeScript, Java, C/C++ |**Backend:** Node.js, Django, REST APIs, Microservices, API Gateway, OAuth2 | **AI/ML**: ML Inference, Multimodal AI, Model Serving, Data Pipelines, Feature Engineering, LLM Systems, Agent-based Systems | **Infrastructure:** Docker, Kubernetes (K8s), Redis, Distributed Systems, Async Processing, Task Queues, Event-driven Architecture | **Databases:** PostgreSQL, SQL, NoSQL, Prisma ORM, AWS S3 | **Frontend:** React, State Management | **Systems:** Linux, TCP/IP, OS Architecture (x86/ARM64/RISC-V) | **Other:** CI/CD, Performance Optimization, Scalability, MCP (Model Context Protocol)