Hongwei Xiao

CONTACT

Phone: +1(213)548-8045
Email: hongweix@usc.edu
Address: 1320 W Jefferson Blvd, LA 90007

PROFESSIONAL EXPERIENCE

C++ Development Intern | 10/2024 - 2/2025

NavInfo Co., Ltd.

- Developed a data matching backend API using Hidden Markov Model to match GPS data with vector maps.
- Applied data normalization to correct path distance distortions from incomplete vector maps
- Building a feature augmentation pipeline to extract missing features from raw vector maps and applying algorithms to ensure over 95% correction accuracy

Penetration Testing Platform Developer | 06/2024-08/2024

Wuhan Qi'an Technology Co., Ltd.

- Built a scalable penetration testing platform with support for diverse targets (web, mobile, firmware, IoT, wireless) leveraging Go, Gin, and GORM.
- Automated vulnerability discovery, validation, and reporting with AI-assisted features, reducing manual workload and accelerating onboarding.
- Delivered extensible and stable solutions through API plugins and optimized high-concurrency integration, improving platform reliability and versatility.

EDUCATION

Huazhong University of Science and Technology | 09/2021-06/2025

B. Eng. in Information Security

- Certified CSP, placed in the top 15% percentile of all test-takers.
- Relevant Courses: Data Structure, Operating System, Compiler Principles, Software Security, Blockchain Technology and Applications, Computer Communication and Networking, Algorithm Design and Analysis

---

SKILLS

- Programming Languages: Python (data analysis, ML, scripting), C/C++, Go
- Frameworks & Libraries: PyTorch, Hugging Face Transformers, scikit-learn, pandas, NumPy, NLTK, Matplotlib; Gin, Zinx, GORM, Vue
- Machine Learning & NLP: Sentence-pair text classification, DeBERTa/RoBERTa fine-tuning, class imbalance handling (WeightedRandomSampler, Focal Loss), learning rate scheduling (cosine decay with warmup), early stopping & gradient clipping, feature fusion (Transformer + MLP), TF-IDF retrieval, Gemma-3 offline inference

- Databases: MySQL, Redis
- Systems & Networking: Linux network programming (epoll, multithreading, IPC, TCP/UDP, HTTP/HTTPS), high-concurrency reactor patterns
- Web Technologies: HTML, JavaScript, CGI
- Tools & Platforms: Hugging Face Hub/CLI, Git, Jupyter Notebook, Docker, CMake, GDB, PowerShell/Bash

---

RESEARCH EXPERIEN

Applying Large Language Models (LLMs) in Algorithm Theory | 02/2025-06/2025

Supervised by Prof. Qiankun Zhang, Huazhong University of Science and Technology

- Supervised by Prof. Qiankun Zhang, Huazhong University of Science and Technology
- Optimized the PASS (EOH) framework with LLMs to tackle NP-complete combinatorial optimization problems using PyTorch.
- Proposed a pipeline-based algorithm generation model, enhancing interpretability through incremental learning steps.
- Improved fitness value in the online bin packing problem from 0.951 to 0.9923 within 17 generations.
- Developed an automated testing pipeline with benchmark datasets, demonstrating superior performance to FunSearch.

SELECTED PROJECTS

High-Concurrency Penetration Testing Server

https://github.com/Ga9a/MyTInyWebServer

- Implemented a high-concurrency network service using epoll and a thread pool architecture.
- Designed an event-driven model with a double-buffering system to support thousands of concurrent connections.
- Integrated a MySQL connection pool to enable efficient database connection reuse.
- Applied modern C++ features (smart pointers, RAII) to ensure memory safety and resource management.

High-Concurrency Penetration Testing Server

https://github.com/Ga9a/MAP-Charting_Student_Misunderstandings

- Designed a 3-stage hierarchy (Gate→Branch→Type) for student misunderstanding detection, achieving Gate macro-F1 ~0.90–0.91 and boosting roberta-base validation F1 from 0.775 → 0.913 with sentence-pair encoding, cosine warmup, and early stopping.
- Implemented rubric-guided offline LLM inference (Gemma-3-1B-IT + TF-IDF) with strict JSON outputs and MCQ ranking fallback, optimized for fully offline deployment.
- Built an end-to-end data cleaning & feature pipeline (LaTeX fraction normalization, explanation cleaning, shallow cues), producing reproducible training curves and confusion matrices.
- Automated Hugging Face publishing (repo creation + folder uploads) with private visibility controls.