Xuzhong Wang

xuzhong7074@gmail.com | +1 8047918290

 github | linkedin

EDUCATION

Bachelor of Science - Computer Science   **Aug 2023 - May 2027**

College of William and Mary, Major GPA 4.0/4.0   *Williamsburg, VA*

Bachelor of Science - Mathematics   **Aug 2023 - May 2027**

College of William and Mary, Major GPA 4.0/4.0   *Williamsburg, VA*

SKILLS

Programming Languages	Python | Java | C++ | C | Go | JavaScript | MATLAB
Technologies	AWS BS/EC/EC2/ELB | Azure AD | Docker | React.js | Pytorch | LangChain | Spring | Git | CUDA

EXPERIENCE

Bestlink Technology   **June 2024 - July 2024**

Software Engineering Intern   *Nanjing, China*

- Developed an innovative automated data retrieval application leveraging **OpenCV** and **Python** to extract and process data from images, boosting productivity of text processing.
- Partnered with senior engineers to architect a robust LLM agent workflow using **LangChain**, enhancing model performance and scalability for advanced AI applications.
- Contributed to the design, development, and training of a cutting-edge LLM utilizing **PyTorch**, optimizing model accuracy and delivering actionable insights.

Scalable Architecture Lab   **May 2024 - now**

Researcher   *Remote*

- Designed and implemented a high-performance data recording tool using **Go** and **SQLite**, enabling efficient and tidy read/write operations to a SQL database for seamless data management.
- Engineered a data mover utilizing a message-passing development pattern in a cycle-based simulation, optimizing data transfer abstraction and system throughput in simulation experiment.
- Implemented a Network-on-Chip (NoC) solution for GPU simulation framwork, integrating **virtual channels** and **traffic classes** to significantly enhance message communication performance.

NOTABLE PROJECTS

Akita Simulator Engine   **May 2024 - Now**

Full Stack Developer   *Remote*

- Spearheaded the development of Akita, a high-performance, modular framework in **Go** for building **computer architecture simulators**, enhancing developer experience through **event-driven design** and Smart Ticking for intuitive **cycle-by-cycle** programming.
- Developed a library of first-party hardware components, including **cache systems, TLBs, memory controllers, and Network-on-Chip** interconnects, accelerating simulator development and enabling scalable, parallelized simulations across multiple CPU cores.
- Supported AkitaRTM, an interactive real-time monitoring dashboard, providing live insights into simulation states and in-flight messages, significantly improving debugging efficiency for issues like deadlocks and performance analysis.

Personal Website   **June 2025 - Aug 2025**

Full Stack Developer   *Williamsburg, VA*

- Designed and developed a personal portfolio website using **HTML** and **markdown** pages, showcasing technical projects and professional achievements with a clean, responsive, and user-friendly interface.
- Implemented a modular and maintainable codebase for the website, leveraging **CSS** and **Jekyll** to create a static appealing platform that effectively communicates technical expertise and project contributions.

Easy21   **May 2025 - June 2018**

Full Stack Developer   *Williamsburg, VA*

- Developed a **model-free reinforcement learning environment** for Easy21, a simplified card game, implementing draw and step functions in Python to simulate game dynamics.
- Applied **Monte-Carlo control** to Easy21, optimizing action-value estimates over 1 million episodes using a time-varying step-size and **ε-greedy exploration**, and visualized the optimal value function to demonstrate effective policy learning.
- Engineered a **linear function approximator** for Easy21 using coarse coding with 36 binary features, integrating Sarsa(λ) with constant exploration and step-size, and conducted comparative analysis of mean-squared error to assess approximation efficacy.