

Yanmei Wang

Phone: (US+1) (734) 548-7207 | E-mail: uyanmei@umich.edu | [Game Gallery](#) | Github: [Yanmeeei](#) | LinkedIn: [Yanmei](#)

EDUCATION

University of Michigan, Ann Arbor, United States <i>M.S. in Computer Science & Engineering, GPA: 3.9/4.0</i>	04/2025
University of Michigan, Ann Arbor, United States <i>B.S. in Computer Science, GPA: 3.8/4.0</i>	04/2023
Shanghai Jiao Tong University, Shanghai, China <i>BSE. in Electrical and Computer Engineering, GPA: 3.6/4.0</i>	08/2023

SKILLS

Languages: C++, C#, Python, HTML/CSS, Swift, GoLang, Javascript

Framework: XCTest, Numpy, Pandas, React.js, Django, Flask, SQLite, Alamofire, OpenAI API

Tools: Git, AWS, Jira, Confluence, XCode, VS Code, JetBrains IDEs, Unity,

Other skills: Agile Development, Data Structures, Unit Testing, Debug, Object-Oriented Design Principles, Relational databases, Web Design & System, Operating System, Software Development Cycle, Machine Learning, Delta Debug

WORK EXPERIENCE

Software Engineer Intern @ PreVeil, Inc. Boston, MA | 06/2024 - 08/2024

- Integrated the PreVeil Drive backend with the **FileProvider framework** on macOS using **GoLang** and **Swift**, successfully mounted the online PreVeil Drive to macOS devices' file system, keeping files updated on all devices.
- Implemented and deployed 22 **RESTful API** endpoints using the **Swift Alamofire** library sending **HTTP** requests, enabling mounting, browsing, and CRUD operations to remote drive directly via macOS Finder.
- Developed and deployed over 30 **unit tests** for the JSON decoder and the backend API endpoints using **XCTest**. Additionally, conducted API testing with **Postman**, ensuring code integrity.
- Iterated and improved **UI/UX** of PreVeil Desktop/Web App using **TypeScript**, **Javascript**, and **HTML/CSS**, improving performance and user experience.
- Engineered a diagnostic API endpoint for PreVeil Drive using **GoLang**, optimizing data pagination from the server backend, which reduced data flow overhead and improved response time by ~1.5x. Improved the corresponding diagnostic frontend webpage using **HTML** and **Javascript** to adapt to pagination.

Software Developer @ Arborsense, Inc. Ann Arbor, MI | 09/2022 - 02/2024

- Implemented multiple data processing techniques using **Numpy**, **Pandas**, and **Scipy** (baseline correction, FFT filtering, etc.) to analyze biological & environmental data with **SQLite**, identifying over 72 types of patient events using **Python**.
- Generated chart reports using **Matplotlib**, enabling the engineering team to visually identify event periods on a timeline, significantly accelerating the development and iteration of algorithms.
- Directed the **code version control** and **documentation** within the engineering team, resulting in a well-organized changelog over 78 iterations to ensure transparency and easy tracking of updates.

RESEARCH EXPERIENCE

Research Assistant @ Clarity Lab (UMich CSE Department) Ann Arbor, MI | 05/2022 - 10/2023

- Designed and implemented the key scheduling algorithm (greedy algorithm) using **Python**. Enables multiple swarm devices to collaborate, achieving up to 3 times speed up for large object detection models under memory constraints. Libraries used: **Numpy**, **Pytorch**, **Scipy**, etc.
- Implemented a simulation-based experimental infrastructure that conducts performance and energy evaluation of the proposed technique/system using **Python**.
- Profiled several neuron network models using **Tegrastats** on low-end edge devices such as NVIDIA Jetson.

PROJECT EXPERIENCE

LLM (GPT-4o) Powered Chatbot 10/2024

- Developed a **Django-based AI chatbot** supporting middle school students' learning experiences by processing video transcripts and providing targeted assistance.
- Performed **prompt engineering** to optimize responses from the assistant according to the student's responses, ensuring that responses were relevant, clear, and age-appropriate for the target audience.
- Developed backend functions in **Python** to register, query, manage, and delete AI assistants through the **OpenAI API**.
- Used **SQLite** database to manage assistants for individual students.

Instagram Clone Web App 10/2022 - 11/2022

- Created an Instagram-like social media website with user profile, posts, comments, likes, and follows features.
- Developed & improved the website layout using **HTML/CSS**.
- Implemented back-end services to update user interactions in real-time using **Python Flask**, and integrated with **SQLite** database for CRUD operations on the user, following, and commenting tables.
- Developed **client-side dynamic pages** with infinite scrolling & updates without refreshing using **React.js**.
- Deployed the application in **AWS EC2** instance.