Hai (Desmond) Zhu

(646) 409-6718 • hz1778@nyu.edu • https://www.linkedin.com/in/hai-zhu-6463021a2 / • https://github.com/dizzyzff

EDUCATION

New York University, Courant Institute of Mathematical Sciences, NY — Dec 2022
Bachelor of Arts, Computer Science — Major GPA: 3.7

TECHNICAL SKILLS

- **Coding Languages:** Python, Java, SQL, HTML, CSS, JavaScript, PHP, C/C++
- **Skills:** Software development, machine learning, data visualization, front-end development
- **Tools:** Git, TensorFlow, SQLite3, MongoDB, Tableau, Datagrip, PyTorch, Keras, OpenCV, ReactJS

EXPERIENCE

Software Engineer Analyst Intern: ***Accenture*** — Feb 2021- Jul 2021

- Part of an engineering team responsible for adapting and consulting new technology for the customers.
- Developed multiple data processing tools, to automate and aggregate the personnel working data weekly, using Python and Pandas, which saved many man-hours on data processing.
- Participated in the overall design of the connecting project. Planned out the project end to end, including overall timelines, network architecture diagrams, documentation, testing, and rollout. After delivering the project, ongoing support of systems by overseeing the evaluation and results from assessments.
- Cross-org collaboration with Accenture's India and United States IT departments, and the Asia Pulp & Paper IT department, by gathering requests and feedback, giving technical support, and holding cross-departmental meetings.
- Delivered the project to customers' IT departments, resulting in cost-saving and improvements in overall efficiency.

PROJECTS

WebDev 2022 ***(JavaScript/CSS/HTML/PHP, SQLite3)*** — Spring 2022

- Hosting the self-made website on the school domain: https://i6.cims.nyu.edu/~hz1778/webdev/index.html.
- Improved the website security by using AJAX requests, hashing user credentials, and storing them in a database outside the public folder.
- Implemented user login status by setting PHP session cookies.
- Built a user chat room with user name functionality and chat history, implemented with PHP, AJAX calls, and SQLite3.

Spotify Music Genre Analysis ***(Python/Data Visualization/ Machine Learning)*** — Spring 2022

- Maintained a training entry database using SQLite to better manage the data and easier to query.
- Applied principal component analysis to data at a massive scale (>25000 data entries) to transform it from a high-dimensional space into a low-dimensional space.
- Built a feedforward neural network model with 10 hidden layers, 10 hidden cells per layer, and a **ReLU** activation function that predicts the genre of Spotify music based on previous data.
- Visualized data by plotting 2D, and 3D PCA plots, K-means Clusters, and the ROC curve of the model.
- Optimized model to fit massive-scale data with an AUC score of 97 percent.

Informational Healthcare Chatbot ***(Python/ Natural Language Processing)*** — Fall 2021

- Implemented a user-friendly GUI for the customer to chat with.
- Retrieved data corpus from CDC website, tokenized training corpus with nltk's WordNetLemmatizer.
- Created and trained a ReLU and sequential neural network model to predict user inputs' classification.
- Applied agile methodology, and managed a small team to meet capstones by planning and holding weekly meetings. Presented to a class of 70 on the approach to building the chatbot and how it performs.

Cyberpunk mini-game (GameMaker Language or GML/Object-Oriented Programing) — Fall 2021

- Designed and drew a modern GUI for the gaming interface, which connects to the gaming backend.
- Designed a multi-level game logic that is fun and challenging, implemented in GML. Organized the code following object-oriented programing principles for maintainability.
- As a major improvement added a multiplayer mode to allow 2 players to play at a time.
- Implemented a game history functionality that outputs and saves every gameplay when it is done.

Simple Disk-based File System and Weensy OS ***(C/ Infrastructure)*** — Spring 2021

- Implemented multiple components of the FUSE driver: allocating disk blocks, mapping file offsets to disk blocks, and freeing disk blocks allocated in inodes.
- Improved the file system to make it crash-resilient, and optimized to make inodes saving more compact.
- Implemented memory isolation, full virtual memory, and fork function for WeensyOS processes.Applied process isolation by giving each process its independent page table.