George Doujaiji

863-303-9369 | GeorgeD.me | Orlando, Florida | GeorgeDoujaiji88@gmail.com
LinkedIn.com/in/george-doujaiji/ | GitHub.com/GeorgeD88

EDUCATION

Oregon State University | B.S. in Computer Science (GPA: 3.8) *April 2023 - Present*

- Relevant coursework: Software Engineering 1, Web Development, Computer Architecture and Assembly Language, Calculus 3, Discrete Mathematics

University of Central Florida | Computer Science *Jun 2020 - Dec 2022*

- Relevant coursework: Data Structures and Algorithms, Object-Oriented Programming, Calculus 2, Systems Software, Intro to Programming with C, Computer Logic & Organization

Certifications

- Supervised Machine Learning by Stanford Online (Coursera)
- Intro to Machine Learning (Kaggle)
- Intro to SQL (Kaggle)

SKILLS

Advanced: Python, Data Structures and Algorithms, Shell/Bash
Proficient: API, Git, Unix/Linux, Object-Oriented Programming, SSH/Tunneling
Familiar: Machine Learning, Deep Learning, TensorFlow, SQL, CI, SCRUM, Web scraping, C
Interpersonal: Communication, Teamwork, Analytical, Inquisitive, English, Arabic, French (novice)

WORK EXPERIENCE

Programming Tutor at Wyzant.com *Sep 2020 - May 2021*

- Delivered ongoing personalized 1-on-1 Python sessions to 11 students, online and in-person, improving their understanding of Python, which is reflected in my 4.9-star rating.

PROJECTS

Shape Classifier Convolutional Neural Network

- Built and trained my first neural network with TensorFlow and convolutional layers for computer vision. Accurately identifies drawings of basic shapes with 98.6% test accuracy.

Movie Recommendation System

- Used content-based filtering to make a system that recommends movies based on semantic content similarity of the movie's title, genres, and tags.
- Leveraged Word2Vec embeddings and used NLP techniques for preprocessing text data.

Minesweeper-Solver

- Recreated the classic game of Minesweeper in Python, engineering an intuitive GUI.
- Developed a bot by designing algorithms using graph theory principles, effectively emulating human gameplay, solving 100% of deterministically solvable boards.

Spotify Trees

- Python-based tool leveraging the Spotify API to manage and organize playlists in a tree structure via a daily Cron job. Successfully handles over 3,000 songs across ~80 playlists.

Pathfinding Visualizer

- Interactive GUI where users can draw a traversable map by placing barriers, start node, and target node, then visualize well known pathfinding algorithms such as A* and Dijkstra's.