YIHANG CAI

365-777-7782 • yihang.cai.nz@gmail.com •

SUMMARY

Third year student of University of Toronto, double majoring in Computer Science and Statistics with a minor in Economics. awarded high distinction for 1st and 2nd year.

EDUCATION

Honours Bachelor of Science, Major: Computer Science, Statistics, Minor: Economics　　　Graduating May 2025
University of Toronto, Toronto, ON
Woodsworth college
Relevant coursework: Hardware Design Languages and Programmable Logic, Advanced Excel in Business
Relevant courses taken: Introduction to Computer programming, An Introduction to Probability and Modelling, Intro to Comp Sci, Math Expr Rsng for Cs, Advanced Calculus, Probability and Statistics I, Probability and Statistics II, Introduction to Applied Statistics, Linear Algebra I

TECHNICAL SKILLS

Data Analysis and Statistics: R studio, SQL, Python pandas
Design and Modeling Tools: Logism, Microsoft Office
Programming: Python, Java

EXPERIENCES

Shenzhen Institute of Advanced Technology (SIAT), Shenzhen, China: Faculty of Computer Science, Center for High Performance Computing　　　May 2023 – July 2023
- Project: Graph Transformer-based Prediction of Drug Target Interactions
- Deepened understanding of machine learning, neural networks, and transformer models, enabling insights into the functioning of Natural Language Processing (NLP) and Large Language Models (LLM).
- Streamlined computational workflows and enhanced project efficiency through the optimization of Linux (Ubuntu) system setups.

ACADEMIC PROJECTS

Data Analysis and Visualization Project　　　March 2023 – April 2023
- Conducted data analysis on CSV files using Python's Pandas library, focusing on reading and manipulating data within data frame objects.
- Applied QQ plot analysis to assess if datasets followed specific distributions and identified deviations.
- Visualized data through plotting histograms and probability density functions (PDFs) to illustrate data characteristics and distribution adherence.

TreeMap Visualization Framework in Python　　　February 2023 - March 2023
- Defining the skeleton for trees, capable of representing both folders (internal nodes) and files (leaves).
- Implemented methods to update, expand, collapse, and visualize the tree structures dynamically, responding to user interaction and visual cues.
- Incorporated logic to allocate screen space proportionally based on the relative data size of the nodes and leaves.
- Implemented functionality for traversal to fetch tree nodes based on their positions, resolving positional ties by choosing the node closer to the origin.
- Created a versatile framework that can be potentially integrated with GUI libraries like Pygame to provide visual insights into hierarchical data structures like file systems.