Yutao ZHOU

Tel: +1 (805)6371617 | **Web:** https://yutao-zhou.github.io/CV/ | **LinkedIn:** **https://www.linkedin.com/in/yutao-zhou/**|**Email:** 13520759678@163.com

EDUCATION

Columbia University — **New York City, NY**
MS in Electrical Engineering — *Sep 2022 – Present*

University of California – Santa Barbara — **Santa Barbara, CA**
BS in Physics — *Sep 2018 – Dec 2021*
Overall GPA: 3.67/4.00
Dean's Honors List in 2021 Winter

SKILLS & SPECIALTIES

Languages: Python (Proficiently used in all of my internship and personal project)
Computer Skills: HTML, MySQL, Shell script, MATLAB, Latex, Auto Desk CAD, EasyEDA, MS Office series

PERSONAL PROJECTS

Used Car Data Visualization Web Application — **Beijing, China**
Independent Projects — *Jun 2022 – Present*

- Build a web application from the idea to rent a domain and host my web application.
- Build with streamlit, handle large data set(365K data points) with **Desk**, **Pandas**, and **NumPy** for data filtering and cache data.
- Visualize data with scatter plot on map, heat map, pie chart, scatter plot with the trend line, using packages like **plotly**, **leafmap**, **pydeck**.
- Geo encoding and filter data with distance from user query location with geoencoder for **GeoPy**.
- Login page with cookie.
- Hosting my web application on a personal server with domain re-direction.
- Successfully used google map API with **request** for geoencoding but abandoned because of the cost.
- Successfully deployed my website to the google cloud platform with **docker** and **yaml** files but abandoned it because of the daily limit of the outgoing internet traffic for the app engine with the free account.
- **Link to my web application: www.auto-showroom.com login with username: guest password: guest**

WORK EXPERIENCE

Deepchem Co., Ltd. — **Beijing, CN**
Python Intern — *June 2019 – Present*

- Group project: create calculation task distribution systems. Work with the front end, and other co-walkers to create a web-based platform that would distribute calculation jobs from the distribution server to different calculation servers based on calculation servers status (Thread usage) and the number of threads that the user declared when submitting the task. I am in charge of all the work after the calculation job started. More specifically the following works:
  - All of the jobs below were performed accordingly to different calculation types, which include ORCA, Gaussian, VASP, and Lammps.
  - Use **GET** to check job status on the platform and handle manual stop from the user. While including a series of actions include stop job in **SLURM**.
  - Use **GET** and **POST** to check front-end job status and submit log content from calculation to the distribution server in real-time.
  - Zip needed calculation results and use **POST** to upload files to the distribution server.
  - Handle a series of situations including but not limited to:
    - Unexpected termination of calculation on calculation server
    - Manual stop or delete from the user
    - Manual cancel job from the server
    - Missing calculation file(calculation files were uploaded from the user when they submit a job)

- The calculation did not start correctly and was terminated right after being submitted with SLURM.
- Job distribution server offline
- Continuity when my program was accidentally stopped

- All of the calculating jobs will back to normal as soon as restart my program. If there are calculation tasks that had been completed they will be uploaded to the distribution server as normal and the log will appear as normal. If there were a new calculation job it will be recognized and tracked. If there are calculation job that was submitted and completed while my program was not running their result will be uploaded to the distribution server normally as soon as my program start. If there are job cancellations they will be handled as soon as my program restart. If the job distribution server is offline, all of the calculation jobs will continue, and if they are finished the result will be uploaded as soon as the distribution server is back online.

- Individual project: writing script that finds missing data set in the database from id queries CSV. Data filtering and aligning. Extract 3D Cartesian coordinate and get SMILE with **Pybel**(OpenBabel) python package. Create and maintain **SQL database**. Extract data from XYZ file, CSV file, and converted SMILE and insert it into SQL database(including checking repeating data in database).
- Individual project: creating automatically data preparation algorithm for ORCA(A calculation software in chemistry) on the server.
- Individual project: creating automatically data collection algorithm from five files we got from ORCA calculation on the server. Including data synchronization from different files and error handling. The algorithm will try to solve the problem when an error is detected.
- Individual project: writing script that collects specified data from a user-specified folder.
- Individual project: attempting to write a script that converts FUNSMILE(our company's way of naming compound structure) to SMILE(a general chemistry structure naming). The company decided to stop this project after a week.
- Individual project: writing script that could communicate with chemistry experiment equipment remotely in the lab and control them to experiment. I used **SOCKET** and designed an algorithm that could check two-way connections automatically.
- Individual project: testing Psi4 calculation speed on the server with different threads and memory settings. Did polyfit to find mathematical minimum time and test hypothesis and wrote reports.
- Individual project: writing script that filters and classifies files. ZIP the chosen file type and upload them to the server with POST with login and token.

Research Institute of Nanjing Runnan Medical Electronics Co., Ltd. **Remote**

Part-time Analytical Assistant *June 2021 – July 2021*

- Collected clinical data in Python and drew signal waveforms, such as electrocardiogram (ECG), electromechanical film ballistocardiogram (BCG), LC BCG, etc.
- Identified the signal peak through local maximization, calculated the peak distance, and measured the beat-by-beat cardiac cycle of ECG signal through plotting the waveform
- Extracted the beat-by-beat cardiac cycle of any BCG signal with or without the synchronous ECG signal reference

ACADEMIC PROJECTS

Auroral Morphology Classification Based on Unsupervised Clustering **Beijing, CN**

Research Project at National Space Science Center, CAS (Advisor: Prof. Ziming Zou) *Aug 2019 – Sep 2019*

- Familiarized with the unsupervised clustering algorithm, KNN algorithm, and K-means algorithm
- Learned about calling the underlying algorithm in the SkLearn machine learning library
- Consulted literature on the morphological categories of aurora observation images in the whole sky and collected the data sources of aurora images
- Programmed to realize morphological clustering of aurora images