Kebing Liu

kebing-liu-personal-website.cyclic.app

🏛 3A Data Science at University of Waterloo | 📍 336 Batavia Place, Waterloo, ON N2L 3W2

📱 365-888-5527 | ✉ k322liu@uwaterloo.ca | 🔗 Kebing Liu | 🐙 github.com/AEsir777

PROGRAMMING SKILLS

Languages: Python, JavaScript, SQL, C++, R, Bash, HTML, CSS
Development Technologies: Node.JS, Express, MangoDB, NPM, JQuery, React, AWS, Git, REST API
Machine Learning Technologies: TensorFlow, Scikit-Learn, Anaconda

WORKING EXPERIENCE

- **Deep Learning Developer** — Jan 2023 - Present
  Environment and Climate Change Canada - Canadian Ice Service — *Ottawa, ON*
  - Set up **Python** Machine Learning environment using **Miniconda** in both Windows and **Linux** Science Network .
  - Designed, implemented the data cleansing pipeline to manipulate Satellite images and improved the efficiency and accuracy of the process of mapping images and SIGRID analysis charts using **Pandas** and **SQLite** library.
  - Computed distance to land countour maps using **GDAL** on land mask and statellite images.
  - Trained **CNN** model using **TensorFlow Keras API** to predict the ice thickness based on the satellite images

- **Quality Assurance** — Sept 2022 - Dec 2022
  Tangam System — *Waterloo, ON*
  - Performed manual testing and modified test plans for metrics, UI and **Hangfire** jobs of SODA, a web application that maximizes the optimization for slots, and reported the issues in **GitLab**
  - Automated over 500 test cases using **JavaScript Cypress** framework end-to-end testing upon **jQuery**
  - Assessed database and SQL injection using **SQL3** JS library and **MS SQL Management Studio**
  - Integrated Cypress tests with **GitLab CI**

- **SAP iXp intern-Quality Engineering Automation Developer** — Fall 2021 – Dec 2021
  SAP — *Waterloo, ON*
  - Designed the automation quality engineering process for the database migration and CRUD operations of **SAP HANA DB** using **Python Selenium** framework
  - Implemented and optimized automation process for in Windows, Linux and Mac Os and different webdrivers are used for various of browsers and Reported flaws to **Jira**

PROJECTS

- **Protein Prediction Model** — Python, TensorFlow, Sckit-Learn, Jupyter Notebook
  - Preprocessed the dataset using scaling, encoding, re-sampling, selects **Random Forest** model for binary classification based on ROC AUC and F1, and tune the model.
  - Hosted the model on **Hugging Face**.

- **Personal Website** — Node.JS, MongoDB, Express, BootStrap, NPM, EJS, JQuery
  - Built responsive user interface using **Bootstrap** and **EJS**, **JQuery** to avoid code repetition.
  - Used **Node.JS** and **Express** for backend and access/update information from **MongoDB** using **Mongoose**

- **CC3K game** — C++ in Linux Env
  - Applied **MVC** and **Oberserver** design patterns to create a Chamber Crawler game with **X11** as view, the game class as Controller over multiple classes as the model.
  - **Decorator** and **inheritance** design patterns were introduced to optimize the code for different races, monsters and treasures.

EDUCATION

Candidate for Bachelor of Mathematics, Data Science, University of Waterloo, Waterloo, Ontario, September 2020 present — Faculty Average: 87.68