Jiasheng Wu

Sunnyvale, CA 94089 jwu.sde@gmail.com 213-245-0285 linkedin.com/in/jiashengwu github.com/jiashengwu

EDUCATION

University of Southern California (USC), Los Angeles, CA *Jan 2018 - Dec 2019*
M.S. in Computer Science, 3.62 *(Artificial Intelligence, Machine Learning, Natural Language Processing)*
Zhejiang University of Technology, Hangzhou, China *Sep 2013 - Jun 2017*
B.Eng. in Software Engineering, 3.75 *(Algorithm, Data Structure, Web Tech., Database, Network, Operating System)*

SKILLS

Programming Languages	**Java**, **Python**, SQL, Shell, C/C++, Swift, PHP
Web Development	HTML5, CSS3, JavaScript, jQuery, AJAX, Bootstrap, React, Angular, Node.js, Spring, Maven, Flask
Software & Tools	AWS, GCP, Git, Docker, Selenium, Redis, MongoDB, Tomcat, Hadoop, Spark, RabbitMQ, Kafka
Architecture	RESTful API, Microservice, Object-Oriented Programming, Design Pattern

EXPERIENCES

Software Engineer Intern at USC Information Sciences Institute (ISI) *7 months / Jun 2019 - Dec 2019*
T2WML: Cell-Based Language to Map Tables into Wikidata Records [github.com/usc-isi-i2/t2wml]

- Brought T2WML into practice by building web-based GUI from scratch using **React**, **Bootstrap**, **Webpack**. GUI includes Excel-like table viewer, T2WML editor with grammar check, and interactive mapping result preview panel.
- Designed highly scalable **RESTful APIs**, such as uploading Excel files and applying T2WML onto spreadsheets, documented in **OpenAPI**, and implemented using **Python**, **Flask**. Automated unit testing using **Selenium**.
- Integrated third-party **Social Login** services, and reproduced Overleaf-like file management features using **Redis**. Allowed T2WML to handle multiple user sessions without file conflicts. Laid solid foundation for T2WML release.
- Created **Docker** containers to migrate T2WML, and deployed on ISI's server running **Nginx**. Opened for inner access and utilized as essential tool in Center on Knowledge Graphs to convert Excel files to Wikidata entries.

Wikidata Fuzzy Search of Time Series [kg2018a.isi.edu/fuzzysearch]

- Implemented fuzzy search by estimating semantic similarity between words using **Word Embedding**. Built **Microservices** with **AWS Lambda** to find synonyms of keyword, and to search in ISI's Wikidata mirror.
- Delivered **Responsive Web App** using **React**, **Bootstrap**, **Flask**. Like Google Images, results are displayed as card collection. Card is uplifted when mouse hover, and half-window preview panel can be toggled when click.
- Wrote robust **SPARQL** queries to fetch time series data from Wikidata as **JSON**, plotted as interactive scatter chart using **D3.js**, and embedded into app as result preview. Tested on desktop browsers and mobile devices.
- Deployed app to **AWS Elastic Beanstalk** using **Git**, **EB CLI**. Enforced **HTTPS** and customized domain as ISI's.

Visiting Research Intern at University of California, Davis *3 months / Jul 2016 - Sep 2016*
Team leader of 3D Reconstruction and Measurement of Chronic Wound

- Reproduced **3D Reconstruction** solutions (like SfM, Structured Light) using **MATLAB**, **OpenCV**. Devised and published 3D point cloud acquisition algorithm based on linear laser and homography.
- Developed **3D Rendering** app to display OBJ files on **Android** using **Java**, **OpenGL ES**. Implemented interactive features like move, pinch-to-zoom and ArcBall rotation. Cached series of files in **Multi-Threads** to play animation.

PROJECTS

Artificial Intelligent Courseworks - *AI, Algorithm* *Sep 2018 - Nov 2018*

- Accomplished pathfinding agents using A* algorithm and utility theory respectively.
- Developed an adversarial agent using minimax algorithm with alpha-beta pruning to reduce search space.
- Solved a Constraint Satisfaction Problem (CSP) of airline scheduling using **Backtracking** algorithm.

Chinese AI and Law Challenge (CAIL '18) - *ML, NLP* *May 2018 - Jul 2018*
Given fact descriptions of criminal cases, predicting judgment results, e.g. applicable law articles, charges, and prison terms

- Achieved Chinese **Word Segmentation** with Jieba, and trained **Word2Vec** model with Gensim.
- Generated additional training data for minor labels to improve prediction performance.
- Assembled deep learning models (**TextCNN**, **BiLSTM**) and fusion model, and reached F1 score of 82%.