Haitian Hao

Email: hait.hhao@gmail.com | Homepage: https://hhao08.com/ | Tel: (+86) 186-0013-0347

EDUCATIONAL BACKGROUND

University of Maryland, College Park (UMD), MD 09/2017 – 05/2021
B.S. in Computer Science, B.S. in Mathematics (Double Degree) | Academic Dean's List | Overall GPA: 3.4/4.0
GRE: V: 166 / Q: 166 / AW: 3.0 / Total: 332

Core Coursework: Algorithms | Advanced Data Structures | PHP & JavaScript | Deep Learning | Machine Learning | Data Science | Information Science | Statistical Computing with SAS | Computer Systems | Object-Oriented Programming I & II | Bioinformatic Algorithms, Databases, and Tools | Computer Vision | Computer Networks | Computational Methods | MATLAB Programming | Multivariable Calculus | Applied Linear Algebra | Differential Equations | Applied Probability and Statistics | Discrete Structures

Certificate: Deep Learning Specialization

SKILLS

Programming Languages: C/C++ | Python | Java | MATLAB | R | SAS | VBA | MySQL | Ruby | OCaml | HTML | CSS | JavaScript | PHP
Software: PyCharm | Eclipse | IntelliJ | GitHub | GitLab | Jupyter Notebook | MySQLWorkbench | MS Office
Libraries: Scientific Computing (NumPy, SciPy) | Data Manipulation (Pandas) | Machine Learning (Scikit-Learn, PyTorch, Keras, TensorFlow) | Visualization (Matplotlib, Seaborn)

RESEARCH EXPERIENCE

Optimize and Innovate Algorithms for Estimating Medoids in Large Datasets 09/2020 - 12/2020
Research Assistant, UMD | Advisor: Prof. Thomas Goldstein

- Developed a novel heuristic algorithm for solving the k-medoids problem to allow data clustering with decent robustness to noise and the ability to handle non-numerical values.
- Attained the objective by solving k trivial sub-problems of centrality; proved that the algorithm's time complexity scales with the number of clusters rather than the number of data points.
- Experimentally evaluated the new algorithm against two commonly used algorithms for k-medoids clustering, showing orders-of-magnitude improvement in computational efficiency and noticeable improvement in cluster quality.

Recognize Cell Types Underlying Coronary Artery Disease Across Humans and Mice 07/2020 - 08/2020
Research Assistant, MIT | Advisor: Prof. Manolis Kellis

- Led a four-member team to analyze the possible causes of a specific type of heart disease - coronary artery disease (CAD).
- Applied PCA, KNN and Seurat to uncover relations between human cell types and CAD based on relations across genes and cell types of humans and mice.
- Leveraged various data visualization techniques in R to facilitate pattern extraction and result presentation.
- Rigorously proved the CAD's significant connection with macrophages and fibroblasts; demonstrated we could diagnose CAD by recognizing macrophages and fibroblasts' status.
- **The first-author thesis paper will be presented in EI; available at my homepage.**

Enterprise Service Governance Based on Service Mesh 08/2019 - 06/2020
Research Assistant, Dajia Insurance Group | Advisor: Dr. Tian

- Developed a mechanism to customize service mesh implementation technologies so as to address the challenges associated with the distributed, microservices architectures (MSA) of enterprise applications, including the 1) management of a large number of services, 2) rapid location and troubleshooting of services, and 3) mitigation of performance issues.
- Applied the mechanism to a local financial company, showing the technical feasibility in solving the service governance problems while covering scenarios, issues, and solutions.
- Demonstrated the new method's ability to provide enterprises with convenient service governance functions, reduce business system development costs, improve operation and maintenance efficiency, and provide standards and specifications for enterprise IT governance.
- **A second-author paper presented in the 2020 6th Annual International Conference on Computer Science and Applications [CSA2020]; available at my homepage.**

WORK EXPERIENCE

Software Engineer Intern **| Dajia Insurance Group (Beijing), China** 06/2019 – 08/2019

- Contributed to the design, implementation, test, and deployment of a cloud computing platform in support of the governance of microservices:
  - Leveraged Kubernetes open-source system for automating deployment, scaling, and management of the containerized applications.
  - Designed the architecture of the cloud platform and put forward an initial plan; proposed a four-layer design, including a cloud framework layer, a middle component layer, a microservice architecture layer, and a unified delivery layer.
  - Employed a docker containers-based strategy to facilitate software development and deployment.
  - Coordinated an internal pilot test; demonstrated the product's ability to support analysis, pricing, optimization of various insurance products.

Software Engineer Intern **| AIA Information Technology (Beijing), China** 06/2018 – 08/2018

- Designed and built a financial system based on Oracle database to attract new customers, retain old customers, and improve customer loyalty to increase market share.
- Employed predictive analytics to aid in employee retention for clients, e.g., determining which employees are most at risk for

leaving based on a number of key indicators.
- Employed SQL-based relational database and Hadoop-based unstructured database to enable efficient internal management of HR data.
- Utilized a suite of data visualization tools, e.g., Tableau and Python tools, to facilitate communication between management, HR specialists, and other employees concerning training and career development.

PROJECT

Development of a Laundry Tracking Application (Java) 07/2020 – 11/2020
- Designed, implemented, and tested an Android application in Java to manage laundry tasks, allowing online reservation and status query:
  - Leveraged XML to improve frontend GUI; enabled backend data storage based on Firebase cloud-hosted database system.
  - Leveraged unified modeling language (UML) to guide software design; followed multiple software engineering principles and design patterns to streamline development processes, focusing on efficiency, modularity, extensibility, maintainability, and robustness.
  - Employed an array of software testing techniques to design test cases, which fundamentally enhanced the robustness of individual methods, GUI, and overall software.
  - Designed and implemented a dynamic programming algorithm to optimize dryer/washer assignment following a set of constraints.

Development of a Stock Price Prediction Application (Python) 03/2020 – 05/2020
- Designed, trained, validated, and tested a reinforcement learning (RL) model to predict short- and long-term stock prices:
  - Completed an extensive literature review on Google Scholar and arXiv, focusing on the publications related to algorithmic trading, deep learning, etc.
  - Designed a hybrid model integrating 1) an LSTM network for encoding the time series data and 2) RL with policy gradient and reward-to-go for price trend prediction.
  - Optimized the algorithm by tweaking loss function, state/action space, reward function, experience replay, and normalization.
  - Collected historical market data from Bloomberg, CHFS, and Wind, followed by backtesting and benchmarking the newly implemented strategies against traditional stock price forecast methods, yielding favorable results.
  - Leveraged AWS cloud computing resources and PyTorch library to facilitate the model development and calculation.

Recognition of False Online Product Reviews Based on Deep Learning 12/2019 – 06/2020
***Deep Learning* Course Project, UMD**
- Collected online product reviews from the Amazon website, followed by data pre-processing to remove outliers and reduce noise.
- Segmented the review texts using the Hidden Markov model, forming a corpus, which was in turn vectorized via the CBOW algorithm.
- Built, trained, and validated a CNN model for false review classification.
- Systematically compared the CNN-based model with the traditional SVM-based model, covering various performance indicators; composed a research article for documenting the findings.

***Machine Learning* Course Project, UMD** 09/2019 – 12/2019
- Built a machine learning tool to assess the vulnerabilities a password subjected to:
  - Leveraged PCA and cosine similarity algorithms to identify patterns of frequently used passwords.
  - Built, trained, and validated an array of supervised learning models for classifying password strength levels, covering decision tree classifier, multilayer perceptron, naïve Bayes classifier, and support vector machine.
  - Benchmarked the performance of the machine learning models against the existing password strength checking tools, with favorable result obtained.

House Price Prediction (Python) 07/2019 – 08/2019
- Completed a systematic regression analysis to predict the house price trend in the city of Shanghai:
  - Mined raw data from several online sources, followed by preparing a dataset with 2K+ entries representing aggregate information about 25 features of homes from different suburbs located in Shanghai; key features include air pollutant concentration, crime rate, student-teacher ratio, among many others.
  - Performed data cleansing to remove low-quality entries, reduce feature redundancy using PCA, handle missing data points, inflation correction to monetary value, etc.
  - Conducted exploratory analysis on the dataset and provided observations; leveraged scatterplot and histograms to facilitate pattern extraction; built a correlation matrix to quantify and summarize the relationships between the variables.
  - Defined proper performance metrics and prepared training/testing datasets; built, trained, and validated multiple regressor models for house price prediction using learning curves and complexity curves while examining the bias-variance tradeoff; leveraged cross-validation and grid search for hyperparameter optimization.
  - Discussed the model's sensitivity and applicability.

Implementation of Data Structures (Java) 09/2018 – 01/2020
- Implemented an array of data structures in Java, including BST, Min-Heap, Max-Heap, Linked List, AVL-Tree, Red-Black Tree, etc.
- Improved speed and memory usage of data structures and algorithms.