JIANFENG CHEN

http://www4.ncsu.edu/~jchen37

919-457-2034 ⋄ jchen37@ncsu.edu ⋄ github: `ginfung` ⋄ linkedin: `janfengcs`

Simplified models can be efficient & beautiful. Software systems should not be too complex. Future software engineers should reduce the system complexities by combining automated SE and machine learning.

EDUCATION

Doctor of Philosophy in Computer Science   Aug 2014 - Dec 2018 (expected)
North Carolina State University, GPA: 3.97/4.0
Coursework: Data Mining | DevOps | Advanced AI | Algorithm Analysis | Graph Theory | Automated SE

Bachelor of Engineering in Computer Science   Sep 2010 - May 2014
Shandong University, China, GPA: 91.1/100
Coursework: Data Structure | OS | Networking | Database System | numerical analysis | Image Processing

SKILLS AND STRENGTHS

Languages	***Proficient:*** Python | Java | LaTeX; ***Familiar:*** JavaScript | MatLab | SQL | C/C++
Data Analysis Tools	Scikit-learn, SciPy, jMetal, Gephi
DevOps Tools	Jenkins, Ansible, Travis-CI, AWS Elasticsearch, S3, Docker

INTERNS AND PROJECTS

(Intern) Brahms Model Verification with Java Pathfinder platform   May 2016 - Aug 2016
Intern in Google Summer of Code program 2016

· Accepted by Google `GSoC2016` program among 18,981 applicants **(accept rate: 6%)**.
· Basing on the `JPF` platform, found the most promising sub-state space in the NASA `Brahms` models.

Sampling vs. Searching in Configuring Large Software Systems   Dec 2014 - Present
NSF funded project in RAISE Lab

· Created a fast sampling technique to replace the common MOEA in Search-based Software Engineering.
· By combining SAT solvers with software product lines, found a way to configure large software systems **2000 times faster**.

LACE Data Privatization Tools and its Application   Aug 2016 - Nov 2016
NSA funded project in RAISE Lab

· Implemented the Large-scale Assurance Confidential Environment (LACE) in python; released the code and documents to `python package index (pip)`.
· Applied LACE algorithm to business and medical data sets. Evaluated the privatization by adversarial accuracy gain, association rules and correlation matrix.

Building Movie Recommendation System   Aug 2015 - Dec 2015
"Netflix Prize" completion extension

· Worked in a team of three to build a movies rating prediction and recommendation system.
· Trained from 100 million `Netflix ratings` by Decision Tree, SVM and ANN.
· Deployed the training process into high performance computing (HPC) server by MPI modules.
· Crawled cast, critic reviews from *rotten tomatoes* and classified the movies basing on Jaro-Winkler Distance.

PUBLICATIONS

[1] **Chen, J.**, Nair, V., Krishna, R., and Menzies, T.. Is "Sampling" better than "Evolution" for Search-based Software Engineering.arXiv preprint arXiv:1608.07617 (2016).

[2] Nair, V., Menzies, T., and **Chen, J.**. An (accidental) exploration of alternatives to evolutionary algorithms for SBSE. International Symposium on Search Based Software Engineering, 2016.

[3] Yang, Y., Zeng, W., and **Chen, J.**. Equiareal parameterizations of NURBS surfaces. Graphical Models 76.1 (2014): 43-55.