Wei Fu

⌂ http://weifu.us • ☎ 919-534-5251 • ✉ wfu@ncsu.edu • ⬡ weifoo • ▤ weifoo

EDUCATION

PhD. CS North Carolina State University   May, 2013 - May, 2018 (expected)
GPA: 4.0/4.0

MS. EE Beijing University of Posts and Telecommunications   Sep, 2009 - Apr, 2012
Overall GPA: 83/100, Major GPA: 86/100.

BS. EE Nanjing University of Technology   Sep, 2005 - Jun, 2009
Overall GPA: 88/100, Major GPA: 90/100, Ranking: 5%.

PUBLICATION

Preprint

[1] **Wei Fu**, Vivek Nair, and Tim Menzies. *Why is Differential Evolution Better than Grid Search for Tuning Defect Predictors?*. arXiv preprint arXiv:1609.02613 (2016). ⍐→`http://tiny.cc/wfuDE`

[2] Amritanshu Agrawal,**Wei Fu**, and Tim Menzies.*What is Wrong with Topic Modeling?(and How to Fix it Using Search-based SE)*. arXiv preprint arXiv:1608.08176 (2016). ⍐→`http://tiny.cc/wfuLDA`

Conference

[3] **Wei Fu** and Tim Menzies. *Easy over Hard: A Case Study on Deep Learning.* In Proceedings of 2017 11th Joint Meeting of the European Software Engineering Conference and the ACM SIGSOFT Symposium on the Foundations of Software Engineering (**FSE'17**). ⍐→`http://tiny.cc/wfuDL`

[4] **Wei Fu** and Tim Menzies. *Revisiting Unsupervised Learning for Defect Prediction.* In Proceedings of 2017 11th Joint Meeting of the European Software Engineering Conference and the ACM SIGSOFT Symposium on the Foundations of Software Engineering (**FSE'17**). ⍐→`http://tiny.cc/wfuOneWay`

[5] Rahul Krishna, Tim Menzies, and **Wei Fu**. *Too Much Automation? the Bellwether Effect and Its Implications for Transfer Learning.* In Proceedings of the 31st IEEE/ACM International Conference on Automated Software Engineering (**ASE'16**). ⍐→`http://tiny.cc/wfuBellwether`

[6] **Wei Fu**, Ruochen Yao, Feifei Gao, James C.F. Li, and Ming Lei. *Robust Null-Space Based Interference Avoiding Scheme for D2D Communication Underlying Cellular Networks.* In Proceedings of 2013 IEEE Wireless Communications and Networking Conference (**WCNC'13**). ⍐→`http://tiny.cc/wfuD2D`

Journal

[7] Jaechang Nam, **Wei Fu**, Sunghun Kim, Tim Menzies, Lin Tan. *Heterogeneous Defect Prediction.* IEEE Transactions on Software Engineering (**TSE**), 2017 (accepted). ⍐→`http://tiny.cc/wfuHDP`

[8] **Wei Fu**, Tim Menzies, Xipeng Shen, *Tuning for Software Analytics: is it Really Necessary.* Information and Software Technology (**IST**), 76 (2016): 135-146. ⍐→`http://tiny.cc/wfuTuning`

PATENT

- Feifei Gao, Wei Fu, J.C.F. Li, and Ming Lei, *Null-space Based Robust Interference Mitigation Method for Multiple-antenna D2D Communication System*, 2012, China Patent. ⍐

EXPERIENCE

ABB USCRC, Software Engineering Group   May, 2016 - Aug, 2016
Research Intern, Raleigh, USA

- Cleaned and visualized historical software development data across all software development teams in ABB.
- Applied data mining techniques to build predictive models and improve the quality of software development in ABB.

China Unicom Design Institute co., LTD   Mar, 2011 - Oct, 2011
Intern, Beijing, China

- Conducted 3GPP standards research on Relay technique in LTE-Advanced system.
- Conducted independent research into the relay technology of a network physical layer, mainly focusing on the performance analysis of the Relay network combined with the network coding technique.

PROJECTS

NSF Funded: Search-based Software Engineering Research Sep, 2014 - Present

Research Assistant Under Dr. Tim Menzies, North Carolina State University, USA

For this project, my research topics mainly focus on how to apply AI techniques to help improve software quality and software process. My research question is always: can we do "it" better and faster? I've investigated the following topics:

- **Parameter Tuning for Software Analytics:** Differential evolution (DE) can quickly find tunings that alter detection precision from 0% to 60%; DE can dramatically reduce clustering instability for LDA and it also leads to improved performances for supervised as well as unsupervised learning. See [2, 8].
- **Differential Evolution v.s. Grid Search:** Differential evolution as a parameter tuner is $210X$ faster than grid search to tune random forests on 17 defect prediction data sets with F-Measure. See [1].
- **Supervised v.s. Unsupervised Learning:** Some supervised data is required to prune weaker models when building effort-aware just-in-time defect predictors. See [4].
- **Simple Techniques for Software Analytics:** SVM with differential evolution-based parameter tuning outperforms the deep learning method for the text classification problem on Stack Overflow, which is also 84X faster. See [3].

NSF Funded: Transfer Knowledge between Software Projects Aug, 2015 - Present

Research Assistant Under Dr. Tim Menzies, North Carolina State University, USA

For this project, my research topics focus on how to use historical data/experience from various sources to improve software project quality. Ive investigated the following topics:

- **Heterogeneous Defect Prediction:** Historical data with different metrics from different projects can be used to build software quality models to predict quality of the target project. By using the mathematical models, we identify categories of data sets as few as 50 instances are enough to build a defect prediction model. See [7].
- **Transfer Knowledge across Search-based SE Problems:** Instead of searching from scratch each time, we propose to utilize the past optimized solutions to guide the search for the newly encountered problem. (In progress).

Build a Continuous Delivery Pipeline from Scratch Sep, 2016 - Nov, 2016

Course Project Under Dr. Chris Parnin, North Carolina State University, USA

I worked on this project for the whole semester. Each component is an independent DevOps task.

- **BUILD**: a component that automatically created a build server, which is capable of building a target project in response to commit events, and trigger a post-build task; track and display a history of past builds via http.
- **TEST**: a component that can generate test cases, run unit tests, fuzzing tests, advanced fuzzing tests with genetic algorithms.
- **ANALYSIS**: a component that run existing static analysis tools, like Jlint, to measure coverage and do code analysis.
- **DEPLOY**: a component that has the ability to configure production environment automatically, deploy the application, monitor the deployed application, auto-scale components of production, perform canary release.

SKILLS

- **Software Development**: Python, Java, ASP.NET, JavaScript, SQL.
- **DevOps**: Ansible, Jenkins, Git, Docker, Vagrant, Redis.
- **Data Science**: Scikit-Learn, PyTorch, Weka, Matlab, R.

TEACHING

- **LAB:** ECE220 Foundations of Electrical & Computer Engineering; ECE109 Introduction to Computer Systems; ECE212 Fundamentals of Logic Design.
- **Guest Lectures:** CSC791 Automated Software Engineering; CSC591 Foundation of Software Science.

SERVICE

- **Program Committees**: FAISE'18
- **Conference Reviewer**: FAISE'18, ESEM'17(subreviewer), IEEE ICC'14, IEEE WCNC'13, IEEE ICC'13, WCSP'12.
- **Journal Reviewer**: IST
- **Member**: IEEE, ACM Student Member.