Derek Rodriguez

dwrodri.github.io dwrodri@g.clemson.edu dwrodri

EDUCATION

Northeastern University (NU)
Predoctoral Student in Computer Engineering *2019 - Present*

Clemson University (CU)
B.Sc. in Computer Science *Class of 2019*

- **Scholarships:**Palmetto Fellows Scholarship, Presidential Scholarship
- **Notable Coursework:** Data Science, Technical Writing, Operating Systems, Computer Systems Organization, Computational Intelligence, Theory of Probability, Multivariable Calculus

EXPERIENCE

CU Scalable Computing and Analytics Laboratory (SCALAB)
Research Assistant *May 2017 - Present*

- Currently optimizing data science pipeline using recurrent neural networks for predicting page faults for GPGPU applications
- Co-authored *Maximizing Throughput on Power-Bounded HPC Systems*, published in IEEE CLUSTER 2018.
- Implemented neural network that profiled hardware usage of software programs to identify opportunities for efficient concurrent execution

Georgia Tech Research Institute, CIPHER Lab
SURE Program Intern *May 2018 - July 2018*

- Designed visualizations of malware network activity for Apiary
- Augmented functionality of Apiary, by implementing visualizations designs using Plotly
- Produced poster and presentation for SURE research symposium

TECHNICAL PROFICIENCY

- **Languages:** Python, C, C++, Java, Bash, LaTeX, ARM/i386 Assembly, HTML
- **Applications:** Tableau, Git, CMake,
- **Platforms:** Linux, macOS, Windows 7/8/10

PROJECTS

Directed Independent Study: Deep Learning *December 2018*

- Extended neural-network-driven active noise cancellation system to support file-streaming from disk, vastly decreasing memory footprint for large datasets

Senior Capstone Design Project: Perceptron Server *May 2018*

- Curated and Implemented NVDocker-based stack for executing deep learning workloads at CU School of Computing.

Linear Quad Trees with Level Differences (LQTLD) *October 2017*

- Python implementation of *A Constant-Time Algorithm for Finding Neighbors in Quadtrees* (2009) by Aizawa and Tanaka
- Optimized map storage by >90% and optimized map cell retrieval from $O(n^2)$ to $O(1)$

References provided upon request