

Animesh Narayan Dangwal

✉ animesh@ucsb.edu • 🌐 drholmie.github.io

Education

University of California Santa Barbara, USA

2021 - Present

PhD in Computer Science, Distributed systems and Edge computing

Advised by Dr. Chandra Krintz and Dr. Rich Wolski, CGPA 4.0/4.0

PES University, Bangalore, India

2016 - 2020

Bachelor of Engineering, Computer Science and Engineering

CGPA: 9.01/10

Coursework

Runtime Systems; Distributed Systems; Accelerator Design; Computer Networks; Fuzzing; Program Analysis; Compiler Design; Operating systems; Machine Learning; Augmented/Virtual Reality

Technical Skills

Programming Languages: PYTHON, JAVA, GO, BASH, RUST, C/C++, JAVASCRIPT

Software Tools: TENSORFLOW, PYTORCH, PERF, EBPf, KUBERNETES, KAFKA,

Experience

Graduate Student Researcher, RACELab, UC Santa Barbara

2021 - Present

- Building accelerator aware and future proof scheduler ecosystems for edge deployments
- Deployments optimally decompose Machine Learning and AI operations across edge devices
- Collaborating with Oracle labs on building schedulers for ML/AI serverless computing on heterogeneous cloud environments

Research Intern, Oracle Labs

Summers 2023, 2024

○ *Extending Cloud Serverless Infrastructure for Multi-tenancy*

Summer 2024

- Extended cloud infrastructure processes to be multi-tenant, while introducing an acceptable trade-off with security and resource utilization
- Reduced cost of keeping idle, provisioned serverless machines by improving utilization proportionally to the (number of CPU cores used by multi-tenant clients)/(total number of CPU cores)

○ *Shared Memory for Multi-tenant Serverless Functions*

Summer 2023

- Profiled the impact of shared memory for serverless runtimes with JVM based applications
- With shared memory, improved warm, cold starts by 9% and 29%, and
- Improved memory usage by 1.2x, against native applications in a multi-tenant cloud

Research Assistant, DREAMLabs, Indian Institute of Science

2020 - 2021

- ElfStore: Developed distributed storage and consistency for edge devices such as raspberry pi, nanos, etc
- UltraViolet: Developed large scale docker deployments and network configurations to emulate IoT architectures

Google Summer of Code Student, CERN

May - August 2020

Accepted as Google Summer of Code for CERN. Developed and designed a local replica of JALiEn central services for research and development. Containerised JALiEn, ALICE's grid computing middleware, using docker deployments for developers to perform local testing and development. Report can be found [here](#)

Publications

Energy optimization and deployment planning for edge devices

ACM CF 2024

P. Guan, A. Dangwal, A. Taherkordi, C. Krintz, R. Wolski

Helmet Design to Detect Drowsiness of Workers On-site Using EEG

IIoT-2019

S. R. Dhole, A. Kashyap, A. Dangwal, R. Mohan

Awards and Accolades

- Oracle Labs Collaboration Funding** 2024
Established a research collaboration with Oracle Labs and UCSB, securing funds and hardware resources from Oracle for the duration of the collaboration
- Summer Graduate Student Researcher Fellowship** 2022
Awarded to student researchers of high academic standing for summer research at UCSB
- Academic Excellence Fellowship** 2021
Awarded to students of high academic standing at UCSB
- IEEE CCEM Pre-Conference Workshop, Best Student Project Award** March 2020
Awarded the **Best Student Project** for "Doctor's Scribe", an application to help doctors reduce time spent on reading and writing long reports by recording, summarising and auto-filling reports.
- Code Gladiators IoT Hackathon by Times of India and Jio** June 2019
First runners-up in the Code gladiator hackathon in the IoT theme and received 125000 INR
- Power of Connected Hackathon by Honeywell** October 2018
First place winners and received team prize of 100000 INR
- C.N.R. Rao Scholarship** 2017, 2018
Awarded to the **top 20%** of the entire Computer Science Batch of 2016 - 2020. Won the scholarship in 2017, 2018

Teaching and Mentorship

- ERSP Mentor, Department of Computer Science, UC Santa Barbara** 2021 - 2022
The Early Research Scholars Program (ERSP) is an academic-year, team-based research apprenticeship program that places special emphasis on mentoring undergraduate students in underrepresented groups.
 - Mentored a team of four UCSB Computer Science/Engineering sophomore students at RACELab
 - Guided the design and implementation of a distributed machine learning pipeline to expedite image classification for conservation efforts at Sedgwick Wildlife Reserve, California.
- Teaching Assistant, Department of Computer Science, UC Santa Barbara** 2021 - 2022
 - Teaching assistant for Introduction to Computer Networking, Fall 2021, Spring 2022
 - Teaching assistant for Introduction to Computer Architecture, Winter 2022, Fall 2022, Winter 2023
- Mentor, Microsoft Innovation Labs, PES University** Summers 2019, 2020
Mentored undergraduate interns at Microsoft's on-campus research laboratory.
Responsibilities included: pitching research projects, guiding students through technical challenges, and organizing events, hackathons.
 - **Approximately Private, Privacy Preserving Machine Learning** Summer 2020
Mentored a team of three. Implemented a machine learning architecture based on **Deep Secure**, focusing on speeding up garbled circuits in machine learning with approximations.
 - **Apathetic ML, Distributed Containerized Machine Learning** Summer 2019
Mentored a team of four. Developed vendor-agnostic distributed deployments for machine learning frameworks using Kubernetes and RabbitMQ.

Projects

- Resource aware adaptive scheduling for microservices** March - June 2022
Designed first order approximations for pipelined usage of machine learning workloads over heterogeneous devices and accelerators to showcase hardware aware scheduling improvements. Our linear model estimated throughput and power, having an error rate of 4.58%.
- Scalable privacy preserving federated learning** October - December 2021
Modified federated learning algorithm with novel load balancing mechanisms to scale deployments and speed up privacy preserving protocols. The system handles asynchronous clients and can accommodate slow devices, reduces time to converge by 20% and network transfer by 30%.