Yile Gu

Address : 2306 Leslie Circle, Ann Arbor, MI

Mobile : +1 734-881-5477

Email : yilegu@cs.washington.edu / yilegu@umich.edu

Personal Website : https://ikace.github.io/

EDUCATION

University of Washington, Seattle, WA

Ph.D. in Computer Science and Engineering Sep 2023 - Present
Advisor: Prof. Baris Kasikci, Prof. Tom Anderson

University of Michigan, Ann Arbor, MI

M.S.E. in Computer Science and Engineering, Cumulative GPA: 4.00/4.00 Aug 2021 - May 2023
B.S.E. in Computer Science, Cumulative GPA: 3.96/4.00 Aug 2019 - May 2021
Award: James B. Angell Scholar, EECS Scholar, Dean's List
Coursework: Compiler Construction, Advanced Operating System, Distributed System, Advanced Computer Vision

Shanghai Jiao Tong University, Shanghai, China

B.E. in Electrical and Computer Engineering, Cumulative GPA: 3.82/4.00, Rank: 11/253 Sep 2017 - Aug 2021
Award: Outstanding Graduate, Merit Student, Undergraduate Excellent Scholarship (Grade 2)(Top 10%)
Coursework: Programming & Data Structures, Intro to Signals & Systems, Intro to Logic Design, Electronic Circuits

RESEARCH EXPERIENCE

Symbiotic Lab, University of Michigan Aug 2022 - Present

Project: Energy Scheduling in Large Model Training *Supervisor: Prof. Mosharaf Chowdhury*

- Discovered the existence of energy bloat in large model training caused by fundamental computation imbalance, where GPUs waste energy by running unnecessarily faster than the critical path of the computation.
- Represented training schedule as a directed acyclic graph (DAG), and designed a graph cut-based algorithm that exclusively and efficiently enumerates all energy schedules on the "iteration time-energy" Pareto frontier.
- Evaluated on large models including GPT3 that our system reduces energy consumption by up to 24% without any slowdown in training time, with only a 30-minutes average running time for the algorithm.

Project: Privacy-enhancing Federated Learning (FL) Platform

- Identified issues with existing FL platforms which lack fundamental support for privacy accounting under different workloads and various types of heterogeneity.
- Containerized core components of an FL platform using Docker for flexible deployment to different operating systems.
- Built a Kubernetes-based coordinator to enable load-balancing support and handle simulated and real FL workloads.
- Designing a privacy-accounting client selector that maximizes FL job utility while respecting clients' privacy budget.

Efes Lab, University of Michigan Jan 2022 - Present

Project: Persistent-memory (PM) Crash-consistency Bug Detection *Supervisor: Prof. Baris Kasikci*

- Spearheaded the development of a PM bug detection tool to address current issues with sub-optimal testing space.
- Leveraged redundancy in PM programs' update behaviors to build dependency graphs for pruning testing space.
- Extended Pmemcheck to trace Linux syscalls and designed an algorithm to test systems with hybrid file operations.
- Assessed the efficacy of the tool on Heterogeneous-memory Storage Engine leading to two PM-related bug discoveries.
- Built an optimized exhaustive testing baseline to demonstrate that our tool can achieve a 32x crash-state reduction.

WORK EXPERIENCE

ByteDance Ltd, Shanghai, China May 2020 – Aug 2020

Software Engineering Intern

- Contributed to a cross-platform mobile application framework with native UI features using C++ and Objective-C.
- Detected and resolved performance bugs in the framework, including a serious memory leak due to circular reference.
- Developed customized components with improved efficiency in rendering logic for mobile application developers.

PROFESSIONAL SERVICE

- **Artifact Evaluation Committee:** OSDI 2023, ATC 2023
- **Student Volunteer:** NSF NeTS PI Meeting 2023

Selected Projects

Enabling Loop Fusion in LLVM by Moving Intervening Code, University of Michigan
- Observed that current LLVM implementation of loop fusion, a powerful compiler optimization that enables better loop distribution and software pipelining, requires unnecessarily strict matching criteria.
- Designed an algorithm attempting to move intervening code before loop fusion if two loop candidates are not adjacent by analyzing data dependencies and determining the correct location for the intervening code.
- Evaluated on microbenchmarks that our algorithm achieves a 40% reduction in running time and a 12% decrease in dynamic instructions executed on average.

Understanding Data Privacy and Byzantine Resilience in Distributed ML, University of Michigan
- Observed theoretical upper bound for combining data privacy and Byzantine resilience with batch size as a bottleneck.
- Determined that a large batch size is required for the convergence of CNN models under Gaussian noise injection.
- Applied gradient sparsification for privacy amplification to account for the fundamental privacy-utility trade-off.
- Discovered that batch size directly correlates with attackers' ability to reconstruct individual images from gradients.

Decaf Compiler, University of Michigan
- Built a lexical analysis scanner using Flex as well as a syntax analysis parser that generates AST based on Bison.
- Implemented a semantic analyzer that performs scope and type checking and supports single class inheritance.
- Created a code-generator for TAC instructions with a register allocator that constructs CFG for liveness analysis and uses Chaitin's algorithm on the interference graph for the k-coloring problem.

Teaching

GSI of Foundation of Computer Science, University of Michigan	Jan 2022 – May 2022 & Aug 2022 - Dec 2022
IA of Academic Writing II and Fantasy Literature, UM-SJTU Joint Institute	Feb 2019 - Aug 2019

Skills

- **Programming:** C++, Python, JavaScript, SQL   **Markup Languages:** HTML, LaTeX, Markdown