Pranay Shrestha

Graduate Student, Computer Science, Stanford University

pranay.shrestha@stanford.edu | stanford.edu/~pranay | github.com/pranay-shrestha | linkedin.com/in/pranay-shrestha

Education

Stanford University, School of Engineering and Technology, *MS in CS* — Sept. 2021 – June 2023

Specialization: Distributed Systems

Relevant courses: Cloud Computing, Operating Systems, Machine Learning Systems

Stanford University, School of Engineering, *BS in CS* — Sept. 2017 – June 2021

Minor: Mathematics — Graduated with Distinction

Publications

Shrestha, P., Lee, J., & Kumar, R. (2022). Adaptive Load Balancing for Geo-Distributed Key-Value Stores. *Proceedings of the ACM Symposium on Cloud Computing (SoCC)*, 14(3), 201–215.

Shrestha, P., & Lee, J. (2021). Gossip-Based Membership Protocols for Large Clusters. *Proceedings of the International Conference on Distributed Systems (ICDS)*, 9(2), 88–97.

Experience

Software Engineering Intern, Google, Mountain View, CA — June 2022 – Sept. 2022

Designed and implemented a service-level objective (SLO) monitoring dashboard for internal storage services, used by 40+ engineering teams to track latency and error budgets.

Wrote a custom alerting pipeline in Go that reduced false-positive pages by 35% by correlating signals across multiple regions before firing.

Graduate Research Assistant, Stanford University, Stanford, CA — Jan. 2021 – June 2022

Advised by Prof. Christos Kozyrakis on distributed storage systems for machine learning workloads.

Built a prototype key-value store in Rust that co-locates model shards with the GPUs that consume them, cutting p99 fetch latency by 2.1×.

Undergraduate Researcher, Stanford University, Stanford, CA — Sept. 2019 – Dec. 2020

Worked with Prof. Dan Boneh on applied cryptography projects, including a study of timing side channels in TLS implementations.

Co-authored a workshop paper on detecting padding-oracle vulnerabilities in open-source libraries.

Teaching

Head TA, CS 149: Parallel Computing — Autumn 2022

Led a team of 12 TAs for a course of 180 students; redesigned the assignment sequence to add a GPU programming unit.

Held weekly office hours and ran a Piazza response-time initiative that cut median wait time from 6 hours to 45 minutes.

Projects

Raft-KV — *Go, gRPC, Docker*

A fault-tolerant key-value store built on the Raft consensus protocol, with snapshotting, dynamic membership changes, and a Jepsen-style linearizability test suite.

Implemented leader election, log replication, and persistent storage; passes 1,000+ randomized test runs without violations.

MiniFS — *Rust, FUSE, Linux*

A user-space file system that stores data in memory and supports basic POSIX operations.

Wrote a write-up explaining the design trade-offs, which was adopted as course material for the following year.

Trace-Viz — *TypeScript, React, D3*

A web-based visualizer for distributed system traces, with support for flame graphs and service-dependency maps.

Open-source project with 300+ GitHub stars and contributions from 5 external developers.

Skills

Languages: Rust, Go, Python, C++, TypeScript, SQL

Systems: Linux, Docker, Kubernetes, gRPC, Raft, Paxos, FUSE

Tools: Git, Bazel, Prometheus, Grafana, Jepsen

Awards and Service

Recipient, Stanford Graduate Fellowship — Spring 2021

Reviewer, *ACM Symposium on Cloud Computing (SoCC)* — 2022, 2023

[illegible]

[illegible]

[illegible]