# Bowen Song

1400 S Figueroa St – Los Angeles, CA 90015 – USA
+1 (857) 318-5369 • bowenson@usc.edu • bowenislandsong.github.io
songbowen • Bowenislandsong • 0000-0002-5071-3880

## Summary

**CS PhD (USC)** specializing in **ML for time series / wearables**, **low-latency inference**, and **large-scale systems**. Industry experience at **eBay (Core Ranking, Infra)** and **Red Hat (OpenShift)**. Published in *IEEE T-BME* and *IEEE CLOUD*. Strengths: **interpretable models**, **GPU training pipelines**, **search/ranking**, and **distributed systems**.

## Education

**University of Southern California** — **Los Angeles, CA**
*Ph.D. in Computer Science* — *2020–2026 (expected)*
Research: ML for Health (wearable sensing, time-series prediction), explainability, and queueing theory for cloud systems.
Advisor: *Leana Golubchik*.

**Boston University** — **Boston, MA**
*M.S. in Electrical & Computer Engineering* — *2019*

**University of Victoria** — **Victoria, BC**
*B.Eng. in Electrical Engineering* — *2017*

## Experience: Health AI & Ranking

**eBay Inc.** — **San Jose, CA**
*PhD Research Intern, **Core Ranking*** — *May–Aug 2024*

- Built and evaluated a **Mixture-of-Experts** (MoE) / **Mixture-of-Rankers** framework for organic search; optimized multi-objective signals (e.g., conversions, clicks, watch metrics) and explored natural **traffic segmentation** and **expert capacity** strategies.
- **Architected** a scalable training pipeline on enterprise **Krylov** across multi-GPU clusters; added **metric monitoring** and experiment tracking for reliable iteration at scale.

**University of Southern California** — **Los Angeles, CA**
*Graduate Researcher, **Personalized AI for Health (QEDLAB)*** — *Aug 2020–Present*

- Developed **interpretable, low-latency** models for biomechanical signal estimation, such as **IMU** → **GRF** and joint signals; designed cross-modal prediction pipelines for athlete **injury risk** monitoring.
- Implemented robust **sensor synchronization** (lag/drift correction) and event alignment for multi-sensor time series.
- Proposed a **linear** personalized biomechanical signal estimator that **competes with** LSTM (published in *IEEE TBME 2024*).
- Designed and evaluated a **Federated** estimator framework that balances global generalization with **local personalization and task / environment specialization**; introduced gating/mixture strategies for unseen tasks across athletes and actions.
- Mentored students; shipped preprocessing & modeling code paths used across projects.

## Experience: Cloud & Distributed Systems

**eBay Inc.** — **San Jose, CA**
*PhD Research Intern, **Infrastructure*** — *May–Aug 2023*

- Owned **Kafka Connector** integration into a multi-datacenter **ClickHouse** store with real-time ingestion; ensured **exactly-once** semantics and streamlined upgrades across environments.

**Red Hat (IBM)** — **Boston, MA**
*Software Engineer, **OpenShift*** — *Jun 2019–Aug 2020*

- Maintained **Operator Lifecycle Manager** and related projects enabling reliable operator installs, dependency resolution, and versioning on Kubernetes.
- Led **Helm** chart integration for operator packaging (import/export of manifests); contributed to upstream **Kubernetes** scheduler/descheduler.
- Built **flake test analyzer** (statistics-driven alarms) adopted in CI to separate flaky from critical failures.

**USC** — **Los Angeles, CA**
*Researcher, **Cloud Resource Management (QEDLAB)*** — *2020–2022*

- Modeled **hybrid-cloud federations** with queueing theory; analyzed performance/cost/QoS trade-offs and revenue sharing (published in *IEEE CLOUD 2022*).

## Selected Publications

B. Song *et al.*, "Estimating Ground Reaction Forces from Inertial Sensors," *IEEE Trans. Biomedical Engineering*, 2024.

B. Song, M. Paolieri, L. Golubchik, "Performance and Revenue Analysis of Hybrid Cloud Federations with QoS Requirements," *IEEE CLOUD*, 2022.

B. Song, A. Trachtenberg, "Scalable String Reconciliation by Recursive Content-Dependent Shingling," *Allerton*, 2019.

R. Liu, B. Song *et al.*, "Contextual Mixture of Ranking Models," *under review*, 2025.

## Skills

**ML**: PyTorch, TensorFlow, CUDA, ranking & MoE/MoR, time-series modeling, evaluation/ablation, reproducible pipelines

**Data/Infra**: Kubernetes, Containers, **Kafka**, **ClickHouse**, Ray, Kubeflow; metric/experiment platforms; GPU clusters

**Languages**: **Python**, Go, C/C++, Java, Scala, MATLAB, R; LaTeX

## Mentoring & Teaching Experience

**University of Southern California** — **Los Angeles, CA**

*Graduate Mentor* — *2020–2025*

- Supervised and mentored **10+ MS and undergraduate students** on projects in ML for biomechanics, cloud simulation, and distributed systems.
- Guided students through end-to-end pipelines: data preprocessing, model design, training, and evaluation; several projects led to **publishable results** and **conference submissions**.
- Developed lab onboarding tutorials for sensor data handling, federated ML, and hand built GPU cluster (now a Lab standard).

**University of Southern California** — **Los Angeles, CA**

*Teaching Assistant* — *2020–2023*

- TA for **CSCI 353 (Computer Networking)**, **CSCI 356 (Computer Systems)**, and **CSCI 402 (Operating Systems)**.
- Led weekly discussion sections and held office hours for **100+ students**, focusing on networking protocols, systems programming, and OS internals.
- Built and maintained an **auto-grading system** for exams and labs, cutting manual grading effort by 60%.

**Boston University** — **Boston, MA**

*Teaching Assistant* — *2018–2019*

- TA for EC528 (Cloud Computing) and EC601 (Product Design). Designed projects, held recitations, and supported students' semester-long team projects.

## Open Source & Selected Projects

**Operator Lifecycle Manager**: Core contributor to the Operator Framework, used widely across the Kubernetes ecosystem for operator packaging and lifecycle management.

**Flake Analyzer**: Built statistical flake test analyzer adopted in CI at Red Hat to detect and classify flaky tests vs critical regressions.

**RCDS Protocol**: Developed the Recursive Content-Dependent Shingling protocol (RCDS) for efficient distributed file synchronization; outperforms `rsync` for frequent updates on extremely large files.

## Service & Community Involvement

**Reviewer**: ACM SIGMETRICS, IEEE CLOUD, *Sensors* journal.

**Conference**: Presenter at IEEE CLOUD 2022; Poster/demo at DevConf.US 2020.

**Organizing**: Co-hosted Red Hat's 2020 Summit (20k+ attendees).