

Sanvi Kaushik

✉️ sanvikaushik@cmail.carleton.ca | LinkedIn | GitHub | (613) 805-6702

TECHNICAL SKILLS

Languages: C++. Python, TypeScript, JavaScript, Kotlin, C, Java, SQL, Bash

Backend & Systems: Node.js, REST/gRPC services, Postgres, Redis, scikit-learn, Pandas

Mobile & Cloud: Docker, Kubernetes, GitLab CI/CD, GCP, React.js

EDUCATION

Carleton University

Bachelor of Engineering (BEng) in Software Engineering

Ottawa, ON

Expected Apr. 2028

- **Awards:** Dean's Honour List, President's Scholarship (\$12,000)

GPA: 3.73/4.0

EXPERIENCE

Software Developer Intern

Ericsson

May 2025 – Present

Ottawa, ON

- Built a **retrieval-augmented chatbot** for **Cloud RAN** documentation, benchmarked across LLMs, and deployed to an internal tool used by **20+ developers**
- Implemented **C++ protocol encoding** for beamforming modules, passing **200+ CI pipelines** to support a major **5G virtualization upgrade**
- Created **20+ Google Tests** validating message encode/decode, increasing pre-release unit-test coverage to **100%**

Software Developer Intern

BlackBerry

May 2024 – Aug. 2024

Ottawa, ON

- Engineered a **C++ synthetic sensor** in the **BlackBerry IVY®** cloud to process vehicle telemetry, improving dog-safety detection accuracy by **40%**
- Trained and deployed a **TensorFlow Lite** model for seat-occupancy detection with **92% precision**, integrating inference into the Android application using **Kotlin**
- Refactored **GitLab CI/CD pipelines** using **YAML** and **Docker**, modularizing SDK tests and stabilizing **100+** builds with **30% fewer failures**

PROJECTS

Trace (1st Place @ Tech-Nol-Hack 2025) GitHub

- Developed a **risk-aware routing engine** computing safer detours in **<200 ms**, raising routing safety scoring by **32%**
- Designed a **privacy-preserving reporter** using anonymous clustering, improving data reliability by **45%**
- Awarded **1st Place** and \$450 prize by a **Microsoft Security Engineer** for measurable civic impact

Global Trade ETL Pipelines GitHub

- Engineered a **parallel ETL pipeline** processing **10K+ XML records/batch**, reducing parse time from **90s → 15s**
- Applied **Pandera validation** for **100% schema coverage**, blocking **455 malformed rows (23%)**
- Implemented **idempotent upserts** with audit logging, sustaining **1.2M rows/min** without duplicates

Dual Wordle Solver GitHub

- Created a **Bayesian solver** achieving **5.43 average guesses** across **300 simulations**
- Built a **Ruby REST API** enabling adaptive inference in **<100 ms**
- Developed a **React UI** visualizing probability updates, reducing user guess count to **7 avg.**