GitHub: Muhammedyakubu
LinkedIn: muhammedyakubu

MUHAMMED YAKUBU

(647) 807-0367
m.yakubu@mail.utoronto.ca

EDUCATION

Toronto, ON **University of Toronto** **Sep 2020 – Jun 2025**

- B.A.Sc. in Electrical and Computer Engineering with Minor in Artificial Intelligence Engineering. **CGPA: 3.85/4.0.**
- Undergraduate Coursework: DSA, Computer Networks, Operating Systems, Deep Learning, Distributed Systems.
- 1 of 4 Recipients of RBC's Award for Diversity and Innovation in Technology - $30,000.

LANGUAGES AND TECHNOLOGIES

- Languages: C/C++, Python, Golang, Java, Verilog, MATLAB, JavaScript, TypeScript, CSS, HTML, TCL.
- Frameworks: React, Django, PyTorch, Flask, Bootstrap, Alexa Skills Kit SDK, Express.js, Angular, Spring Boot, JUnit.
- Technologies: Node.js, Jenkins, MongoDB, Git, FPGA, Arduino, Protocol Buffers, Socket.io, OSM, Pandas, NumPy.

EMPLOYMENT

Software Engineer, Intern **Google** *(Gmail Intelligence Quality)* **May 2024 – Aug 2024**

- Improved the quality of SmartLabel (Gmail's inbox sorter) training data, as measured by reduced human-evaluator bias, by designing and implementing a Python pipeline for cross-evaluation. Ensured user privacy and data security by removing Personally Identifiable Information (PII) from donated emails before cross-evaluation in the pipeline.
- Enabled business-critical insights into consumer behavior, guiding the development of an upcoming Gmail product, by building a distributed C++ pipeline that uses LLMs to securely analyze outgoing emails

ML Compiler Engineer, Intern **Cerebras Systems** *(ML Compiler Backend)* **Oct 2023 – May 2024**

- Migrated our compiler backend to LLVM, enabling separate compilation of Cerebras-Assembly and C++ source files, improving ML kernel development flexibility and simplifying supporting for future hardware generations.
- Wrote algorithm to compress TableGen files, cutting size by 26%, line count by 18%, and generation time by 12%.
- Implemented scoping, register allocation and other language features using advanced data structures & algorithms
- Developed a large volume of high-performance C++, learned about Cerebras' weight-streaming paradigm and utilization of MLIR & SSA to optimize graph-compilation of entire ML models to their wafer-scale engine (SoC).

Embedded Software Engineer, Intern **Apple** *(watchOS & iOS Display Drivers)* **Jul 2023 – Sep 2023**

- Designed and implemented a DMA controller driver feature (in C++) which streamlines PIO register programming and unlocks new avenues for enhanced performance through wider feature adoption.
- This enhancement reduced lines of code by 95% for each application, significantly improving feature usability.
- Debugged intricate Hardware/Software interactions involving IO-MMU and embedded co-processors on the SoC.
- Identified and resolved an existing memory allocation bug, reducing feature memory usage by 62.5%

Software Engineer, Intern **Google** *(Memorystore for Redis)* **May 2023 – Jul 2023**

- Enhanced the team's benchmarking framework (in Golang) with storage, visualization, analysis, and alerting features, ensuring the detection of performance regressions and boosting the rollout reliability of our product.
- Authored a 12-page design document outlining major decisions and future extensions to the project.
- Developed a library to improve scalability, resulting in a 90% reduction in test configuration time and lines of code.
- Reduced implementation time by 50% (4 weeks) by researching, proposing and adopting alternative internal tools.

Software Developer, Intern **Royal Bank of Canada** *(Consumer Banking)* **May 2022 – Aug 2022**

- Isolated the batch-running component from RBC's client onboarding platform, resulting in 10x faster build times.
- Upgraded the batch-runner's functionality using Java's Spring Boot Framework and deployed it to Cloud Foundry via an automated Jenkins pipeline, enabling non-real-time activity refreshes for over 1,000,000 clients daily.
- Overhauled a major Angular component to make it more reusable, reducing its content update time by 80%.

ENGINEERING DESIGN TEAMS

Co-founder **Syllabyte** **Feb 2024 – Present**

- Building a product which provides students with a daily, automatically prioritized to-do list for their coursework, aimed at students who struggle with time-management (implemented in React & Django).
- Designed and implemented the system to extract course deadlines from a syllabus using a Large Language Model and prompt tuning, allowing users to setup their entire semester in minutes.

Software Engineer & PM **Distributed Systems Project** **Jan 2023 – Apr 2023**

- Led a 3-member team to design and develop a highly available, dynamically scalable, fault-tolerant distributed key-value store (in Java) with a caching layer for sub-millisecond latency, employing JUnit for test-driven development.
- Enhanced the storage service with a notification mechanism, enabling client subscription to data mutations. This allows for improved latency and scalability while enabling event-driven workflows and analytics.

Machine Learning Engineer **Deep Learning Projects** **Sep 2023 – Dec 2023**

- **Pokémon classifier**: Worked on a team of 4 to develop and train a model which utilizes transfer learning with a

pre-trained CNN (GoogLeNet) and a custom ANN to accurately classify 800+ Pokémon into 18 distinct types.
- Employed data cleaning and augmentation techniques (linear transformations, guassian noise) using pandas and numpy to enhance dataset quality and quantity, while fine-tuning hyperparameters with stochastic gradient descent to optimize model performance and prevent overfitting.
- **Spam detector**: Implemented and trained a recurrent neural network to detect spam emails with 97% accuracy.

Software Engineer & PM **LastMile GIS** **Jan 2022 – Apr 2022**
- Managed an agile team of 3 students to develop a Geographic Information System (in C++) with searching and pathfinding functionality, using the UnitTest++ framework to support test-driven development.
- Implemented stateful graphics and multi-threaded Dijkstra's algorithm, reducing map load times by 60%.
- Created a route planning algorithm to make parcel deliveries more efficient and cost-effective.

Firmware Engineer **University of Toronto Formula Racing Team** **Dec 2021 – July 2022**
- Wrote and tested software (C++) which ran safety checks on the analog signals from our car's pedals and relayed the corresponding throttle value to the inverter, enhancing driver safety and pedal responsiveness.
- Developed a pin driver for the rear micro-controller of the car to provide a programming interface for the components that communicate with it, hence accelerating development time and improving code portability.
- Created, tested, and calibrated a C++ module to report battery temperatures to the battery management system.

Quantitative Analyst (C++) **Quantitative Impact Investment Club** **Sep 2020 – Jan 2021**
- Worked on the Quantitative Research Team to explore strategies related to Quantitative Momentum.
- Designed and Implemented a High-Frequency trading market maker algorithm in C++ & Python.

COMMUNITY & LEADERSHIP

Finance Director **National Society of Black Engineers (NSBE)** **Oct 2021 – June 2023**
- Managed a team of 4 students: scheduled and conducted meetings, delegated tasks, and created status reports.
- Worked on a team of 11 executives to organize NSBE's annual hackathon – NSBEHacks (300 participants).
- Revised and managed a $13,000 budget (NSBEHacks), cutting costs by 40% from the previous year.
- Networked with industry professionals to acquire corporate sponsors, increasing YoY revenue by 20%.

PROJECTS

Asteroids – C **Apr 2022**
- Recreated an enhanced version of the popular 1979 game entirely from scratch with multiplayer capability.
- Wrote a low-level graphics interface, which enabled the game to run on Intel's DE1-SoC board via its VGA output.
- Refactored the graphics interface and implemented double buffering, boosting in-game performance by 10x.

ChatRoom - HTML, CSS, JavaScript, Socket.io, Express.js & Node.js **Dec 2021 – Feb 2022**
- Developed a web-based group-chatting application with a minimalist front-end featuring a quick login mechanism.
- Created a REST API - hosted separately from the front-end - with Mongoose, Express and Node.js