


# TORY C

☎ (+1) 623-309-3519 ✉ [t376chen@uwaterloo.ca](mailto:t376chen@uwaterloo.ca)  [linkedin.com/in/torychen/](https://www.linkedin.com/in/torychen/)

## Education

### University of Waterloo

Sep 2021 – Apr 2026

Bachelor of Computer Science — Dean's Honours

Waterloo, ON

- **Overall Average:** 93.2%, **GPA:** 3.97/4.00
- B.P. Dammizio Entrance Scholarship, President Scholar of Distinction

**Relevant Coursework:** Data Structures and Algorithms, Object-Oriented Programming, Application Development, Operating Systems, Compilers, Machine Learning, Statistical Analysis, Linear Algebra, Calculus, Numerical Computation

## Technical Skills

**Languages:** Python, TypeScript, JavaScript, Java, C++, C, SQL, Bash, HTML, CSS

**Technologies:** Flask, sklearn, LangChain, React, Node.js, Express.js, Spring Boot, Kafka, Flink, REST APIs, NoSQL

**Developer Tools:** Git, Linux, AWS, GCP, Kubernetes, Docker, Webpack, Postman, VS Code, Jupyter Notebook

## Experience

### X (Twitter) | Python, TypeScript, React, GraphQL

Jan 2025 – Apr 2025

Incoming Software Engineer

Palo Alto, CA

- Will be joining X's internship program as a Software Engineer Intern on the Monetization team and will utilize modern web APIs and frameworks, to craft client components and features that will shape the future of development of X.com.

### IBM | Java, Python, TypeScript, Spring Boot, Kafka, Flink, MongoDB, Kubernetes

May 2024 – Aug 2024

Software Developer

Toronto, ON

- Designed a test coordinator system that automates regression testing for **350+** Kafka Streams apps on a **Kubernetes** cluster using a microservice architecture that employs a **LLM** for schema mapping and **Flink Jobs** written in **Java** for data generation and validation.
- Created a dynamic and responsive user interface for the test coordinator system that allowed the user to select the test app, edit test values, view test results and leveraged **caching** to reduce load times using **React** and **TypeScript**.
- Built a **Kafka Streams** app that generates **110 000+** messages and deployed it on a **Kubernetes** cluster using **Helm**.
- Automated the matching of test scripts using **vector embeddings** and the **Reciprocal Rank Fusion** algorithm, saving **5 mins** per script.

### IBM | Python, LangChain, Flask, Neo4j, AWS, GCP, Docker

Jan 2024 – Apr 2024

Machine Learning Developer

Toronto, ON

- Improved the accuracy of **RAG systems** for **LLMs** by **65%** by replacing the **vector database** with a **Neo4j** graph database that is integrated with **vector embeddings** and stores data chunks based on structure and hierarchy.
- Built a parser API that generates a hierarchical graph structure from .pdf and .docx files using **Python** and **Flask**.
- Developed a **LangChain** retriever and reduced **Neo4j** query times by **63%** through optimizing the **Cypher** queries.
- Automated **Java** code conversion from **Struts 2** to **Spring Boot** using the **Claude model** integrated with **RAG**.

### Dr. Inbox | Python, pandas, DynamoDB, AWS

May 2023 – Aug 2023

Software Developer

Cambridge, ON

- Created **CI/CD pipelines** on **AWS** that automate **unit testing**, **integration testing** and deployment for serverless Lambda functions and led the adaptation of **IaC** by provisioning cloud resources using **Cloudformation** and **SAM**.
- Improved accuracy for automated prescription refills by **38%** through fuzzy matching and enhanced brute force strategies.
- Developed a Python GUI script that syncs Dr. Inbox's data with the PS Suite EMR using **DynamoDB**, **SNS** and **SQS**, saving **30 mins** daily.

### YuJa | TypeScript, React, Express.js, Webpack, SQL, AWS

May 2022 – Aug 2022

Software Developer

San Jose, CA

- Built a custom synchronized webcam and screen video player that replays videos stored on **S3** using **React** and **Video.js**.
- Designed the **SQL** database schema that supports **230 000+** student users and wrote queries that monitored video processing, validated account data and authenticated user sessions using the **Sequelize ORM**.
- Improved video processing efficiency by **87%** by leveraging **Node Worker threads** to process videos concurrently.

## Projects

### Markdown Editor | TypeScript, React, Material UI

- Built a web based **Markdown** editor that compiles **Markdown** into **HTML** for display using **React** and **TypeScript**.
- Developed a **lexer** that scans the input and converts it into tokens by using a variant of the **maximal munch** algorithm.
- Implemented a **parser** that changes the lexer tokens into a valid **syntax tree** and used the tree to generate HTML.

## Hobbies and Interests

- Soccer, Badminton, Swimming, Climbing, Poker, Chess, Hiking, Cooking