

# Sunjay Varma

[sunjay.ca](http://sunjay.ca) [sunjay03](https://twitter.com/sunjay03)  
[varma.sunjay@gmail.com](mailto:varma.sunjay@gmail.com)

[github.com/sunjay](https://github.com/sunjay)  
[linkedin.com/in/sunjayv](https://linkedin.com/in/sunjayv)

Proven, exceptional skills refined by over 10 years of experience writing high quality software

## Publications & Speaking Engagements

### **Intelligent and Affectively Aligned Evaluation of Online Health Information for Older Adults** - [sunjay.ca/aaai2017](http://sunjay.ca/aaai2017)

Julie M Robillard, Areej Alhothali, Sunjay Varma and Jesse Hoey  
AAAI Workshop on Health Intelligence, San Francisco, CA, 2017

### **StarCon 2018 - How Rust Makes Advanced Type Systems Accessible to the Masses** - [starcon.io](http://starcon.io)

Making a case for advanced type systems and the Rust programming language

## Work Experience

### **Compilers Research Assistant** - [plg.uwaterloo.ca/~cforall](http://plg.uwaterloo.ca/~cforall)

Programming Languages Group, Waterloo, ON, Sep 2017 - present

- Implemented a dynamically-sized, copy-on-write string type with its own internally managed string heap and memory allocator
- Investigated bugs in the cforall parser (the C grammar is non-trivial)

### **Artificial Intelligence Research Assistant** - [chil.uwaterloo.ca](http://chil.uwaterloo.ca)

Computational Health Informatics Lab, Waterloo, ON, Sep 2016 - Sep 2017

- Applied Natural Language Processing (NLP) and Machine Learning (ML) techniques to rate the quality of online health articles
- Rapidly learned NLP and ML topics necessary to produce results without any prior background knowledge
- Authored tools to aid in data collection, analysis, and machine learning

### **Software Developer** - [konradgroup.com](http://konradgroup.com)

Konrad Group, Toronto, ON, Jan - Dec 2016

- Led the planning and implementation of the entire frontend and backend for a project from start to finish

### **Lead Software Architect & Developer** - [janetingley.com/anyware/](http://janetingley.com/anyware/)

anyWare (University of Waterloo), Waterloo, ON, May - Aug 2015

- Led a small team in the design and development of the entire software architecture for orchestrating several robotic sculptures
- Reliably synchronized multiple software-based and robotic clients in real-time with high volumes of input

### **Intermediate Web Software Developer Co-op** - [smartamp.com](http://smartamp.com)

SMART Technologies, Calgary, AB, Sep - Dec 2014

- Performed at or above the level of an intermediate developer, mentoring interns and authoring features

### **Web Software Developer Co-op** - [smartamp.com](http://smartamp.com)

SMART Technologies, Calgary, AB, Jan - Apr 2014

### **Full-stack Developer Part-time** - [clinicsense.com](http://clinicsense.com)

Clinic Sense, Toronto, ON, Jul 2013 - Oct 2014

## Technical Overview

### **Programming Languages:**

Rust, ES6, JavaScript, Node.js, C++, Haskell, C, Python, HTML, CSS, SCSS, Bash, SQL, and more

**See page 2 for my extensive open source portfolio**

## Education

### **Candidate for Bachelor of Software Engineering, Co-op University of Waterloo**

May 2017 - present

### **Computer Engineering, Co-op University of Waterloo**

Sep 2013 - Aug 2016

## Open Source Work & Personal Software Projects

### **Rust Programming Language Compiler** - [github.com/rust-lang/rust](https://github.com/rust-lang/rust)

Rust is a systems programming language that runs blazingly fast, prevents segfaults, and guarantees thread safety

- Mentored by Niko Matsakis (Rust Core Team) on the implementation of RFC 1598 Generic Associated Types
- Extended the parser to support the necessary syntax for generic associated types
- Updated name resolution and lifetime resolution code to support generics in associated types

### **Teaching Programming to Complete Beginners - Turtle Graphics in Rust** - [turtle.rs](https://turtle.rs)

Library for teaching the Rust programming language to beginners with animated drawings

- Writing a course with the goal of teaching people who have never programmed before to write Rust
- Designed a graphics library that makes learning programming engaging, approachable, and fun
- Hid the entire complexity of the library behind a small, well documented, and easy to learn interface
- Fostering a community of contributors and students who help with bugs and add example programs

### **LOGO Educational Programming Language Compiler** - [github.com/sunjay/logoc](https://github.com/sunjay/logoc)

A compiler for the LOGO educational programming language using my Rust turtle graphics library

- Created a compiler that generates valid Rust code based on an input LOGO program
- Rust code is used as an intermediate representation (IR) and compiled into binary using the Rust compiler
- All type checking and error handling is done in the LOGO compiler, Rust is only used for code generation

### **Brain Programming Language Compiler** - [github.com/brain-lang/brain](https://github.com/brain-lang/brain)

A compiler written in Rust that takes a strongly-typed, Rust-like syntax and compiles it into "brainfuck"

Brainfuck is a programming language with only 8 very primitive instructions. Compiling to brainfuck is not easy because you have to very creatively compose its limited instruction set in order to represent complex programs.

- Designed a compiler with the following stages: parsing, static checking and IR generation, static memory layout and instruction generation, optimization, and code generation
- Invented novel algorithms for simulating complex constructs like nested branching, loop control flow, and boolean operators using very few instructions and maintaining assumptions about memory management

### **Fast Brainfuck Interpreter** - [github.com/brain-lang/brainfuck](https://github.com/brain-lang/brainfuck)

A fast, optimizing interpreter for the brainfuck programming language written in Rust

- Authored a complete, in-depth specification of the brainfuck programming language (see GitHub repo)
- Optimized the performance of brainfuck programs by batching instructions while the program is loaded

### **Fast Sudoku Solver** - [github.com/sunjay/sudoku](https://github.com/sunjay/sudoku)

Very fast Sudoku solving algorithm implemented in C

- Optimized algorithm by reducing memory allocation and applying heuristics before naive search methods
- Of the 21,885 puzzles tested, 87.2% were completed in less than 1 second, 81.9% were completed in less than half of a second, and 71.1% were completed in less than 0.1 seconds

For even more projects: [github.com/sunjay](https://github.com/sunjay)

For my writings on various technical topics: [sunjay.ca](https://sunjay.ca)