

# Spandan Garg

spgarg@microsoft.com

in/glGarg/  
github.com/glGarg

## EDUCATION

---

- **Bachelor of Software Engineering, University of Waterloo** Waterloo, ON  
*Notable Courses: Machine Learning, AI, Performance, Databases, Operating Systems* Sept. 2015 – May 2020
- **Machine Learning, Coursera - Stanford University** Online

## TECHNICAL SKILLS SUMMARY

---

- **Languages:** Python, C/C++, C#, Java, HTML, javascript
- **Libraries/Frameworks:** Fairseq, PyTorch, TensorFlow, Keras, scikit-learn, .NET, OpenGL

## WORK EXPERIENCE

---

- **Microsoft Corporation** Sep. 2020 - Present  
*Data & Applied Scientist* Home Office
  - **Performance Bottleneck Detection:** Developed a technique to detect performance bottlenecks in C# and .NET applications from CPU/Memory profiler traces; showcased at the VSLive! Conference (2021).
  - **Performance Pull Request (PR) Collection:** Trained a semi-supervised Random Forest model to gather performance related PRs from Github repos. Used model to collect ~1350 PRs with a wide range of performance improvements.
  - **Performance Improvement Suggestion:** Developed a change suggestion technique by extending Facebook's existing code search technique, Aroma, by adding performance specific features to suggest improvements for detected bottlenecks, using collected performance PRs.
  - **Performance Patch Generation (Ongoing):** Collected all C# repos with >5 stars on Github and crawled commit history to generate examples with various file/class level contextual elements (import statements, class attributes, method signatures, etc.). Finetuned BART (after pre-training on English & Code) for performance patch generation. Further details can't be disclosed due to NDA.
- **Microsoft Corporation** Sep. 2019 - Dec. 2019  
*Data Science Intern* Redmond, WA
  - **Frequent Snippet Extraction from Telemetry:** Extracted short scripts a.k.a snippets of Azure CLI commands from usage telemetry by splitting telemetry into individual usage sessions and finding frequent n-grams of successful commands, to generate concise and intuitive snippets.
  - **Human Language to Snippet Search:** Used FastText word embeddings and Azure documentation to resolve out of vocab (OoV) words, fix typos, perform query expansion, etc. in an incoming query. Developed a novel code search technique using the tree-like hierarchy of Azure CLI commands to translate human language queries to snippets of Azure CLI.
  - **Snippets Extension for Visual Studio Code:** Wrote a VS Code extension demonstrating the snippet search approach ([Demo Video](#)) using Python, Flask and typescript; demoed at Microsoft Azure+AI Conference (2019).
- **Microsoft Corporation** Jan. 2019 - Apr. 2019  
*Data Science Intern* Redmond, WA
  - **Azure CLI Command Example Generation:** Built an AI project to generate examples for Azure-CLI documentation using telemetry. Generated example templates using the most impactful command and parameter combinations in telemetry. Scraped CLI examples from StackOverflow, Github Issues, Blog posts etc. and extracted parameter values to reduce placeholders (instances where we can't generate values for a parameter) in generated examples by >55%.
  - **Parameter Type Classifier:** Trained a machine learning model using Random Forest and GloVe embeddings (trained on Azure documentation) to classify command parameters into types (e.g. String, Integer, Filepath, IP-Address, etc.) and used it to further improve example quality by ensuring type-accurate values are selected during example generation. Pipelined the entire process of data collection, feature engineering and model training and deployment.
  - **Automatic Example Submission Pipeline:** Wrote a pipeline to automate example validation and submission of PRs containing new examples to Azure-CLI Github repo and Azure Docs. Used this pipeline to add examples to the documentation of high-impact Azure CLI modules that receive >96% user request share.

- Microsoft Corporation**

May. 2018 - Aug. 2018

*Software Development Engineer*

Vancouver, BC

    - **Streaming Install Feature:** Implemented streaming install for Gears of War 5 on Xbox using C++, Windows Runtime Library and Xbox XDK. This feature allows users to play the game as it installs.
    - **DLC Install Pipeline:** Implemented Downloadable Content (DLC) installation pipeline for the game using Concurrency Runtime and Xbox API.
  - Capcom Vancouver**

Sep. 2017 - Dec. 2017

*Software Engineer*

Vancouver, BC

    - **Remote Build Tool:** Developed a command line tool to trigger remote builds using Python and TeamCity and integrated it into project's build system and Perforce.
    - Performed tasks like parallelizing build scripts, writing smoketests and collecting build statistics for analysis.
  - Next Level Games Inc.**

Jan. 2017 - Apr. 2017

*Rendering Programmer*

Vancouver, BC

    - **Material Editor Tool:** Worked on a material editor tool using C# and .NET. Implemented various features such as the ability to open multiple tabs at once, file search, joystick control within viewport, etc.
    - **Graphics Regression Detection Tool:** Developed a web-based image comparison tool to visualize game's screenshot comparison data using Mako, Python, HTML and javascript. Implemented a scalable, well-documented solution to enable engineers to easily detect graphics regressions after each build.

## PATENTS AND PUBLICATIONS

---

- Spandan Garg, Paul Harrington, Roshanak Zilouchian Moghaddam, Chen Wu and Neel Sundaresan. 2021. System and Method For Identifying Performance Bottlenecks. Filed November 12th., 2021. Patent Pending.
- Spandan Garg, Roshanak Zilouchian Moghaddam, Chen Wu, and Neel Sundaresan. 2021. PerfLens: A Data-driven Performance Bug Detection and Fix Platform. In *Proceedings of the 10th ACM SIGPLAN International Workshop on the State of the Art in Program Analysis (SOAP'2021)*.
- Spandan Garg, Paul Harrington, Roshanak Zilouchian Moghaddam, and Chen Wu. 2021. Performance Bug Detection and Code Recommendation. Filed March 10th., 2021. Patent Pending.
- Spandan Garg, Roshanak Zilouchian Moghaddam, Yevhen Mohylevskyy and Jason Shaver. 2019. Command-line Script Generation with Relevance Search. U.S. Patent Application 20210342357, Filed May 1st., 2020. Patent Pending.
- Spandan Garg, Roshanak Zilouchian Moghaddam, Jason Shaver and Neel Sundaresan. 2019. Machine Generated Examples of Command-line Commands with Parameter Values. U.S. Patent Application 20210342654, Filed April 29th., 2020. Patent Pending.

## SCHOLARSHIPS AND AWARDS

---

- **University of Waterloo, Department of Math:** Dean's Honours List
- **University of Waterloo, Department of Engineering:** Graduation with Distinction
- **University of Waterloo, Department of Engineering:** President's Scholarship
- **Alberta Education:** Alexander Rutherford Scholarship

HOBBIES: Sketching ([Link To Artwork](#)), Computer Graphics ([Link To Graphics Demos](#)), Running, Badminton