

# Rahul Krishna

http://rkrns.us ◊ Github: github.com/rahlk

Email : i.m.ralk@gmail.com

Mobile : +1-919-396-4143

LinkedIn: linkedin.com/in/rkrns

## Education & Training

### Postdoc in Computer Science

Columbia University

June 2019 – February 2021

### PhD in Computer Science

North Carolina State University

June 2015 – December 2019

### MS in Electrical Engineering

North Carolina State University

August 2013 – May 2015

### BE in Electronics & Communication

Ramaiah Institute Of Technology

August 2009 – May 2013

## Technical Skills

**General Expertise:** Generative AI, Software Engineering, Hybrid Cloud, Machine Learning, Statistics, Natural Language Processing;

**Programming:** Java, Python, C++, Golang, Rust;

## Work Experience

### IBM | Staff Research Scientist

Feb 2021 - current

**Languages:** Java, Python, Golang, Rust; **Tools/Frameworks:** Treesitter, Antlr, WALA, SOOT, Kubernetes, vLLM;

- **LLM-based code translation and test generation:** Currently contributing to the research and development of the WatsonX™ Code Assistant for Java, focusing on the use of Code-LLMs in two critical use cases:
  - (1) *Programming language translation from ABAP to Java:* Constructing and leveraging Code-LLMs along with Retrieval-Augmented Generation (RAG) tailored for code to generate accurate translations of ABAP to Java. This helps the transition from legacy systems on SAP to modern Java frameworks and platform (specifically Liberty, Quarkus, and Openshift).
  - (2) *Automated test generation for Enterprise Java applications:* Enhancing LLM's ability to generate test cases for enterprise Java application with a focus on: (1) Advanced model fine-tuning and alignment techniques; (2) Optimizing model performance through quantization, facilitating on-premise deployment for customers.
- **Mono2Micro and CARGO:** Drove Research and Development for IBM's Mono2Micro, an AI-guided microservice recommendation tool that leverages static and dynamic program analysis for migrating legacy applications to cloud-native architectures. Spearheaded the development and publication of CARGO (see <https://github.com/IBM/codenet-minerva-cargo>), which garnered a *Distinguished Paper Award at ASE 2022*. This partitioning algorithm has been integrated into the Mono2Micro offering, significantly enhancing its capabilities. Mono2Micro and CARGO projects contributed multi-million dollar revenue impact with our global enterprise clients, earning an *O-Level accomplishment* for the efforts.
- **Konveyor Data Gravity Insights (DGI):** Drove Research and Development for Data Gravity Insights (DGI) within the RedHat Konveyor initiative. DGI modernizes enterprise applications by analyzing code, data, with a specific focus on understanding and refactoring database transactions. Project is available at <https://github.com/konveyor/tackle-data-gravity-insights>. This contribution also earned me an *O-Level accomplishment* at IBM for its impact on application modernization.

### Columbia University | Postdoctoral Researcher

June 2019 - Feb 2021

**Languages:** Python, C, C++; **Frameworks:** Tensorflow, Pytorch, LLVM/Clang, NVIDIA Jetson SoC, Hadoop, Spark;

- **Diagnosing and Fixing Performance Faults in Real-time Systems using Causal Inference:** Developed a causal inference based diagnostics and repair tool, called CADET, for diagnosing root-cause of performance faults in real-time embedded systems. CADET's diagnosis outperformed expert recommendations and other state-of-the-art techniques by 28–52% while doing so up to 40× faster.
- **Scalable Performance Optimization for Big Data systems using MCMC:** Developed a Markov Chain Monte Carlo based configuration optimization prototype, called ConEX, for big data systems. Demonstrated that ConEX can *frugally find near optimum configuration settings* for Hadoop and Spark with over 17× better latency gain than competing ML based approaches.
- **Transferable Graybox fuzz testing with Multi-task Neural Networks:** Designed, implemented, and deployed a Multitask Neural Network based fuzz testing framework, called MTFuzz, to expose vulnerabilities in C/C++ programs. MTFuzz creates a *compact embedding* of the program space to find 11 unseen vulnerabilities and increase code coverage by 2–3×.
- **Vulnerability prediction using GGNN and Representation Learning:** Developed a vulnerability prediction tool called REVEAL using GGNN architecture to improve prediction performance in real-world software systems such as Chromium, Debian OS, FFMpeg, and QEMU. REVEAL produces a 33.57% boost in precision and 128.38% boost in recall over state-of-the-art methods.

### Phase Change LLC | Software Engineer Intern

June 2018 - August 2018

**Languages:** Python, Java; **Frameworks:** Tensorflow, NLTK, Scikit-Learn, Gensim, Apache Jena

- **Extracting domain knowledge embedded in COBOL.** Developed an ontology based vector embedding model to automatically summarize financial domain knowledge embedded in monolithic COBOL code. Achieved confidence scores  $\geq 80\%$  on determining banking concepts in COBOL code as measured by surveying subject matter experts and active COBOL developers.

### LexisNexis | Data Scientist Intern

May 2016 - August 2016 | May 2017 - August 2017

**Languages:** Python, Java, Scala; **Frameworks:** Tensorflow, Spark, Livy, AWS-EMR, NLTK;

- **Technology assisted review using Machine Learning:** Developed a sandbox application for automated document retrieval in Technology Assisted Review (TAR) and E-Discovery, leveraging Kernel-SVM and active learning to enhance the recovery of relevant documents from a skewed sample set, achieving  $\approx 20\%$  improved accuracy. Additionally, created a text summarization framework to generate concise “headnotes” for over 1 million legal documents, employing scalable clustering and doc2vec classification to categorize documents into legal topics, effectively reducing document review time by  $\geq 50\%$  as per in-house surveys.

## Selected Publications

(For a full list of publications, see: <https://scholar.google.com/citations?user=WGggocoAAAA>)

★★ Distinguished Paper Award

★★ Best Paper Award runner-up

- [1] Pan, R., Ibrahimzada, A. R., Krishna, Rahul, Sankar, D., Wassi, L. P., Merler, M., Sobolev, B., Pavuluri, R., Sinha, S., and Jabbarvand, R., “Lost in Translation: A Study of Bugs Introduced by Large Language Models while Translating Code” In **Proceedings of the 46th International Conference on Software Engineering (ICSE '24), April 14-19, 2024, Lisbon, Portugal. 2024**. Link: <https://arxiv.org/abs/2308.03109>
- [2] ★★ Nitin, V., Asthana, S., Ray B., Krishna, Rahul, “CARGO: AI-Guided Dependency Analysis for Migrating Monolithic Applications to Microservices Architecture” In **Proceedings of the 37th International Conference of Automated Software Engineering (ASE 2022)**. Link: <https://arxiv.org/pdf/2207.11784.pdf>.
- [3] ★★ Chakraborty, S., Krishna, Rahul, Ding, Y. and Ray, B., “Deep Learning based Vulnerability Detection: Are We There Yet?”. In **IEEE Transactions on Software Engineering (TSE), 2022**. Link: <https://arxiv.org/abs/1703.06218>.
- [4] Iqbal, M S., Krishna, Rahul, Javidian, M., Ray, B., Jamshidi, P., “Unicorn: Reasoning about Configurable System Performance through the lens of Causality” In **EuroSys 2022**. Link: <https://arxiv.org/pdf/2201.08413>
- [5] Kalia, A., Xiao, J., Krishna, Rahul, Sinha, S., Vukovic, M., and Banerjee, D., “Mono2Micro: A practical and effective tool for decomposing monolithic java applications to microservices”. In **IEEE Transactions on Software Engineering (TSE), 2021 (Under review)**. Link: <https://arxiv.org/abs/1703.06218>
- [6] She, D., Krishna, Rahul, Yan, L., Jana, S. and Ray, B., “MTFuzz: Fuzzing with a Multi-Task Neural Network.” In **Intl. Conference on Foundations of Software Engineering (ECSE/FSE), 2020**. Link: <https://arxiv.org/pdf/2005.12392.pdf>
- [7] Krishna, Rahul<sup>†</sup>, Tang, C.<sup>†</sup>, Sullivan, K. and Ray, B., “ConEx: Efficient Exploration of Big Data System Configurations for Better Performance.” In **IEEE Transactions on Software Engineering (TSE), 2020**. Link: <https://arxiv.org/pdf/1910.09644>
- [8] Krishna, Rahul, Nair, V., Jamshidi, P. and Menzies, T., “Whence to Learn? Transferring Knowledge in Configurable Systems using BEETLE.” In **IEEE Transactions on Software Engineering (TSE), 2020**. Link: <https://arxiv.org/abs/1703.06218>
- [9] Krishna, Rahul, and Menzies, T., “Learning actionable analytics from multiple software projects.”. In **Empirical Software Engineering (EMSE), 2020**. Link: <https://arxiv.org/pdf/1708.05442.pdf>
- [10] ★★ Wang, J., Yang, Y., Krishna, Rahul, Menzies, T. & Wang, Q., “Effective Automated Decision Support for Managing Crowdstesting”. In **Proceedings of the 41st International Conference on Foundations of Software Engineering (ICSE '19), 2019,** Link: <https://arxiv.org/pdf/1805.02744.pdf>.
- [11] Krishna, Rahul & Menzies, T., “Bellwethers: A Baseline Method For Transfer Learning”. In **IEEE Transactions on Software Engineering (TSE), 2018**. Link: <https://arxiv.org/abs/1703.06218>
- [12] Krishna, Rahul, Agrawal, A., Rahman, A., Sobran, A., & Menzies, T. “What is the Connection Between Issues, Bugs, and Enhancements? (Lessons Learned from 800+ Software Projects)”. **Intl. Conf. Software Engineering (ICSE), 2018 SEIP**. Link: <https://arxiv.org/abs/1710.08736>
- [13] Chen, D., Fu, W., Krishna, Rahul, & Menzies, T. “Applications of psychological science for actionable analytics”. In **Intl. Conference on Foundations of Software Engineering (ECSE/FSE), 2018**. Link: [arXiv:1803.05067](https://arxiv.org/abs/1803.05067)
- [14] Chen, J., Nair, V., Krishna, Rahul, & Menzies, T. “Sampling as a Baseline Optimizer for Search-based Software Engineering”. In **IEEE Transactions on Software Engineering (TSE), 2018**. Link: [arXiv:1608.07617](https://arxiv.org/abs/1608.07617)
- [15] Krishna, Rahul, Menzies, T., & Layman, L. “Less is more: Minimizing code reorganization using XTREE”. In **Information and Software Technology (IST), 2017**. Link: <https://arxiv.org/abs/1609.03614>
- [16] Krishna, Rahul, Menzies, T., & Fu, W. “Too much automation? The Bellwether Effect and its Implications for Transfer Learning.” In **Intl. Conference on Automated Software Engineering (ASE), 2016**. Link: <https://doi.org/10.1145/2970276.2970339>