

- Software Engineering
- Programming Languages
- Static Analysis
- Security and Reliability

Fifth-year PhD student in Computer Science with a focus on pluggable type systems for detecting null pointer dereferences, security, and automating code fixes using large language models (LLMs).

## EDUCATION

- **Ph.D Student, Computer Science and Engineering**, University of California, Riverside. Advisor: [Prof Manu Sridharan](#) **2020 - Present**
- **B.Sc., Computer Engineering (Software)**, Sharif University of Technology, Tehran, Iran **2014 - 2019**

## PROFESSIONAL EXPERIENCE

**Ph.D Student Research Intern** **2022 Summer**  
**Uber Technologies Inc.**

- Joined Programming Systems Group ([PSG](#))
- Enhanced [NullAway](#), Uber's type-based checker designed to prevent NullPointerExceptions (NPEs) across Java and Android monorepos.
- Developed [Annotator](#), an automated tool for practical inference of nullability types, used by engineers within Uber's Java team.
- Customized Annotator with support for module boundary considerations, enabling seamless integration of NullAway across Uber's monorepo.
- Automated the annotation inference of **millions** of lines of code, significantly reducing manual effort for NullAway compliance.
- The project led to the complete enrollment of Uber's java monorepo into NullAway.
- Published research with Uber at the **Foundations of Software Engineering (FSE) 2023** conference: [Practical Inference of Nullability Types](#)

## PUBLICATIONS

**Practical Inference of Nullability Types** ([pdf](#), [talk](#)) **2023**

- [FSE2023](#) - Nima Karimipour, Justin Pham, Lazaro Clapp, and Manu Sridharan. (In collaboration with **Uber Technologies Inc.**)
- Introduces a tool to automatically infer nullability type qualifiers, easing NPE prevention in large Java codebases.
- Uses a black-box search strategy to maximize type-checked code.
- Achieves a 69.5% reduction in NullAway errors on average across Java projects.
- Successfully deployed in production, Enrolled 160+ modules into NullAway and 1.3 million lines of code internally at Uber.

**Practical Type-Based Taint Checking and Inference** **2024**

- Submitted - Nima Karimipour, Kanak Das, [Behnaz Hassanshahi](#), and Manu Sridharan. (In collaboration with **Oracle Labs**)
- Proposes a new type-based taint checker to reduce false positives, improving handling of third-party libraries and complex language constructs.
- Introduces an automated technique to infer tainting type qualifiers, including support for polymorphic annotations and generic types.
- Outperforms whole-program taint analyzers (e.g. CodeQL), achieving comparable precision and recall while operating 27 times faster.
- Demonstrates effective, source-compatible annotation inference, making the tool practical for real-world deployment.

**A New Approach to Evaluating Nullability Inference Tools** **2024**

- [FSE2025](#) - Nima Karimipour, Erfan Arvan, Martin Kellogg, and Manu Sridharan.
- Identifies bias in prior evaluations of nullability inference tools, showing that "type reconstruction" experiments can inflate effectiveness.
- Proposes a new, unbiased definition of "best" inferred annotations, better suited for comparing inference tools.
- Conducts the first head-to-head comparison of three inference tools, revealing their unique strengths and areas for improvement.

## PROJECTS

**NullAwayAnnotator (Annotator)** **2023**

- Accelerates checker adoption by automating annotation search to minimize errors, simplifying onboarding for existing codebases.
- Instantly generates patches with no remaining checker errors, offering a solution unmatched by other tools.
- Configurable to consider build target boundaries, avoiding annotations that could cause downstream dependency issues.
- Supports large-scale projects incrementally, enabling gradual checker integration target by target.
- Scalable to millions of lines of code, fully integrating NullAway across Uber's entire monorepo.
- **Recognized for its impact**, with a dedicated blog post, [Automating Java Codebase Annotations for Null Safety](#), published by [Gitar](#).

**NullAway** **2022**

- A widely adopted tool to help eliminate NullPointerExceptions (NPEs) in Java code.
- Enhanced to support expressing methods preconditions and postconditions.
- Added a serialization service to log errors and compute quick-fix suggestions for reported errors by the checker.

**TaintChecker** **2024**

- A type-based taint analysis, enabling modular and incremental checking for better scalability and performance.
- Reduces false positives by handling third-party libraries and complex constructs, improving taint analysis precision.
- Automatically infers tainting type qualifiers, including support for polymorphic and generic type argument annotations.
- Outperforms whole-program taint analyzers (e.g. CodeQL) with 27X faster checking time and comparable precision.

## ACADEMIC AWARDS

**Deans' Distinguished Fellowship Award** **2020**

- PhD program at University of California, Riverside